Fedora 7 » GP2X » Scribus DTP » ZFS

LXFDVD

LINUX FORMAT

Главное в мире Linux

Gentoo 2007

ПЛЮС: KDE 4!

Бета-версия на LiveCD

» Более 50 отличных приложений!

4 607130 820551

Сентябрь 2007 № 9(96)

Пытаем Linux новичками с.42

СОЗДАЙТЕ БОЛЕЕ ЧИСТУЮ, БЫСТРУЮ И СТАБИЛЬНУЮ СИСТЕМУ

Linux без шлаков

Освежите и взбодрите свой дистрибутив переустановкой – не теряя ценные данные и настройки! с. 26

Развивайтесь

Linux помогает поднимать IT в развивающихся странах с. 46

Редактируйте

Научитесь обращаться с *Vim*, как ветеран Unix с. 64

Играйте

Создавайте собственные игры для Linux и не только: начала с. 122

«Найдется много талантливых хакеров, которые могут помочь вам с GEGL»

Ойвинд Колас с. 32

Linux center
www.linuxcenter.ru

LINUX FORMAT
Главное в мире Linux

К Вашим услугам...

Поскольку они никогда, никогда не допускали в этом деле ни единой промашки, мы решили поинтересоваться у Команды LXF: «Какой совет вы бы дали человеку, собирающемуся переустановить Linux?»

Пол Хадсон
Выделите по разделу на каждый файл – для максимальной безопасности.

Грэм Моррисон
Поставьте в фоне Так медленно, как только можете Джона Кейджа – и зациклите!

Майк Сондерс
Напечатайте шестнадцатеричный дамп своей файловой системы – так, на всякий случай и, чтобы быть уверенным наверняка, снимите ксерокопию.

Эфрейн Эрнандес-Мендоса
Попробуйте сначала собрать компьютерный столик из IKEA – остальное покажется вам семечками.

Д-р Крис Браун
Развлекайтесь, вынимая клавиши из клавиатуры кнопки и переставляя их в случайном порядке.

Дэвид Картрайт
Читайте сообщения, прежде чем жать на кнопки – делай вы это раньше, Linux не пришлось бы переустанавливать.

Энди Ченнел
Всегда ожидайте, что вы потеряете все, нажитое непосильным трудом – включая семью и доброе имя, и делайте резервные копии.

Энди Хадсон
Ммм.... Совет? Не пробуйте Gentoo.

Нейл Ботвик
Чем меньше времени у вас есть на решение возникающих проблем, тем больше проблем возникнет.

Миссия журнала

- Пропаганда свободного ПО в России
- Продвижение решений с открытым кодом в бизнес-сообществе
- Поддержка российского Open Source сообщества
- Организация трибуны для разработчиков свободного ПО
- Обратная связь между разработчиками и потребителями ПО

Двойной праздник

«Уважаемый читатель! Вы держите...» гхм, где-то я это уже писал. Ну, как бы там ни было, вы держите в руках юбилейный, двадцать пятый выпуск русскоязычного Linux Format, которому в этом месяце исполняется ровно два года. Такой вот двойной праздник получается.

За прошедшее время в журнале многое изменилось. Мы стали плотнее сотрудничать с отечественными авторами и корреспондентами, сменили дизайн, увеличили тираж, запустили электронную версию и даже выложили архив за 2005–2006 год в свободный доступ. Хочется верить, что все эти изменения были к лучшему, но если у вас есть свое мнение на этот счет, мы будем рады услышать его – приходите на выставку *LinuxLand*, приносите пригласительный билет и ищите знакомый синий флаг **LXF**. Ну а если в начале октября вы никак не сможете посетить Москву, не расстраивайтесь: день рождения – не единственный двойной праздник, который мы планируем отмечать в этом году. Не за горами еще один юбилейный, сотый выпуск англоязычного Linux Format и путем несложных арифметических действий вы можете вычислить, на какой месяц он приходится. Так что держите руку на пульсе – дальше будет только интереснее! **LXF**

Валентин Синицын » Главный редактор info@linuxformat.ru

Как с нами связаться

Письма для публикации: letters@linuxformat.ru
Подписка и предыдущие номера: subscribe@linuxformat.ru
Техническая поддержка: answers@linuxformat.ru
Проблемы с дисками: disks@linuxformat.ru
Общие вопросы: info@linuxformat.ru
Web-сайт: www.linuxformat.ru

› **Адрес редакции:** Россия, Санкт-Петербург, ул. Гончарная, 23, офис 54.
› **Телефон редакции:** (812) 717-00-37. Дополнительная информация на стр.128

LINUX FORMAT

Сентябрь 2007 № 9(96)

Содержание

Весь номер – прямо как на ладони: приятного чтения!

Учебники

Jumpcut
Редактируем видео в сети 56
Ваши первые шаги в увлекательный и волнующий мир видеомонтажа.

› **Научись делать кино!**

Bash
Информация о пользователях 60
Базовые команды, которые наверняка пригодятся новичку – фундамент нашей новой серии из четырех уроков о работе в командной строке.

Evolution
Распорядок дня 62
Слишком заняты, чтобы следить за своим временем? *Evolution* управится со всем за минуту!

Vim
Введение в основы 64
Это непростой редактор, но ваши труды будут вознаграждены сторицей.

Mono
Напишем чат-клиент 68
Если вам надоело держать у уха трубку или вы хотите иметь собственную почтовую службу, этот урок – для вас.

APT
Продвинутые трюки 72
Вы научитесь многому и сможете управлять APT на полной скорости и стопроцентной мощности.

Java EE
Общение в Сети 76
Намучились с сокетами в Mono? Посмотрите, как изящно и масштабируемо можно сделать то же самое в Java!

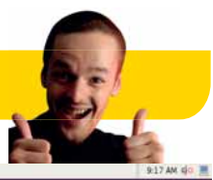
Препринт
Ризограф 80
Вы можете думать, что этот аппарат не сложнее ксерокса – до тех пор, пока не попробуете подготовить что-нибудь для печати на нем.

Cinelerra
Титры 82
Эта серия подходит к концу – самое время показать титры и рассказать о том, как их делать.

RADIUS
Защищаем Wi-Fi сеть 84
Вы узнаете, как поднять безопасность домашней сети до уровня предприятия, а заодно познакомитесь с *FreeRADIUS*.

LXF DVD96

Майк вам покажет 96



Gentoo 2007.0

Хотите иметь полный контроль над своей ОС? Вливайтесь в сообщество Gentoo – вход здесь!

Symphony OS 2007

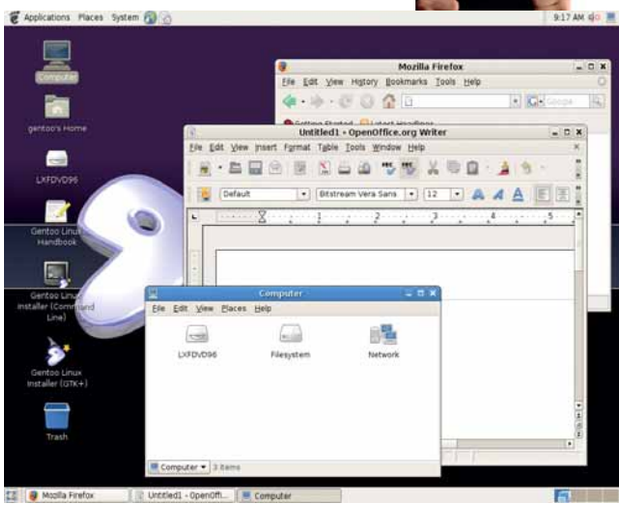
Революция началась! Symphony нравится быть немножко другой – так что располагайтесь поудобнее и загружайтесь. Нам кажется, что эта версия заслуживает внимания.

KDE 4 Beta Live

Оно уже почти здесь – и это куда больше, чем очередной релиз с исправлениями ошибок. Популярное рабочее окружение стало быстрее и ярче.

Свежие релизы

Держите руку на пульсе – вашему вниманию предлагается вкусная коллекция свежих программ и обновлений, включая Gimmage, Nero и Empathy.



› **Gentoo славится своим сообществом. Хотите стать его частью?**

Что за штука... ZFS?

Крупнейшие из современных хранилищ данных — взгляд изнутри с. 52

LXF HotPicks

Лучшие новинки открытого ПО на планете..... 104

› **Загляните в будущее современных дистрибутивов с помощью *KTorrent*.**

LXF ИНТЕРВЬЮ

«Я сильно боялся приниматься за GEGL. Поначалу все было таким непонятным.»

Ойвинд Колас с. 32

Open Source — вокруг света с. 46

Как свободное ПО помогает сократить «цифровую пропасть»

Подпишись на Linux Format и сэкономь!

Содержание

LXF DVD внутри!

Хотите узнать, что на нем? См. с. 96

Спецрепортаж

Linux без шлаков

Переустановите систему по правилам и получите отличный результат с. 26

А также...

Scribus 36
Сверстаем статью с нуля

Linux-тест для новичка 42
Как с ним справятся трое наших «подопытных»?

Вся власть – народам 46
Сообщества всех стран – соединяйтесь!

Обзоры

Defcon 12
Игры, замешанные на мировом господстве и ядерных ракетах, приводят вас в трепет? Наша команда предлагает взглянуть на них с улыбкой.

› **Обозревая *Defcon*, мы учинили разрушения эпических масштабов.**

Fedora 7 14
Сумеет ли очередной релиз Fedora сдвинуть Ubuntu с насиженного места?

GP2X 15
Если вы любите доморощенные игры, это должно вам понравиться.

Nero Linux 3 16
Может ли запись CD и DVD быть простой, быстрой и надежной? Попробуем спросить у Nero.

TwonkyMedia 4.4 27
Поиск мультимедиа-устройств, совместимых с Linux, заканчивается здесь.

Сравнение: почтовые клиенты

Amarok 19
XMMS 20
Juk 20
Rhythmbox 21
BMPx 22
Banshee 22

Постоянные рубрики

Новости 04
В деле SCO поставлена очередная точка.

DistroWatch 24
Мы выберем шесть лучших Linux-инноваций за последние десять лет, а также поговорим о Slackware и играх.

Интервью LXF 32
Человек, который воскресил GEGL, чтобы сделать мир пользователей GIMP чуточку лучше...

Что за штука 52
Последнее слово в разработке файловых систем растянулось на две страницы в нашем журнале.

Мнение 54
Ведущий разработчик Samba рассуждает о бизнес-моделях на примере Стар Трека.

Ответы 88
В этом месяце: печать, Apache, ядро и все остальное.

Школа LXF 110
Букет интервью к началу учебного года

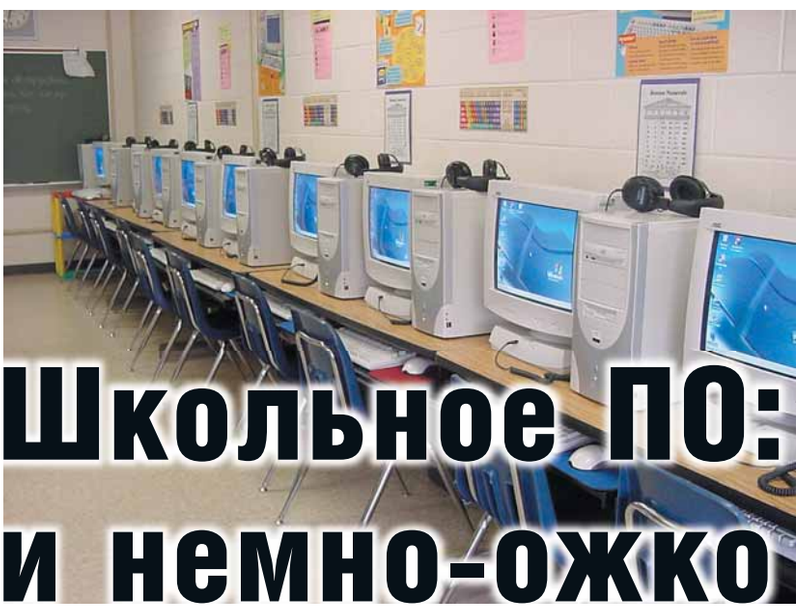
Игрострой 122
Хотите научиться писать игры для Linux? Спросите нас, как.

Через месяц 128
LXF97 – одним глазком

› **Эллисону есть что сказать**

Новости

ГЛАВНЫЕ НОВОСТИ: » В школах пока что останется Windows » OOXML: трудности стандартизации » Linux на персональных компьютерах » Вынесено решение по делу SCO » GNU Affero GPL » Citrix приобретает XenSource



» Рубрику ведет Илья Шпаньков

Школьное ПО: Windows и немно-ожко Linux

Пока школьники догуливают каникулы, готовясь к очередному напряженному учебному году, разработчики ПО развернули нешуточную борьбу за право предоставить учащимся свои продукты. Впрочем, разгоревшиеся страсти довольно быстро улеглись после заявления первого заместителя Председателя Правительства РФ Дмитрия Медведева: он четко дал понять, что вопрос с закупкой лицензий на проприетарное ПО, предложенной Министерством информационных технологий и связи РФ, решен положительно. На эти цели правительство выделяет более трех миллиардов рублей, которые будут постепенно потрачены в течении трех ближайших лет на закупку операционной системы Microsoft Windows, офисного пакета MS Office, а также средств разработки, программ для редактирования графики и антивирусных решений известных коммерческих производителей. При этом не стоит упрекать российские власти в «нелюбви» к свободному ПО – до начала учебного года остались считанные дни, а образовательным учреждениям так и не был представлен сколько-нибудь приемлемый вариант свободной операционной системы, удовлетворяющей требованиям российской системы образования. Справедливости ради нужно отметить, что проприетарное ПО вообще не проходило проверку на пригодность к использованию в школах, но в течении последних пяти лет преподаватели смогли хоть немного приспособить существовавшие на рынке программные продукты под образовательные нужды, поэтому во избежание излишних проблем и было решено легализовать хотя бы то, что уже есть на школьных компьютерах.

Впрочем, не стоит думать, что в сражении за сферу образования свободное ПО понесло безоговорочное поражение. Как и все здравомыслящие люди, российские чиновники прекрасно понимают опасность зависимости от определенных поставщиков ПО (особенно ярко демонстрирует «прелести» монополизма недавнее 25-процентное повышение цен на продукцию Microsoft в России), поэтому одновременно с закупкой для школ проприетарного программного обеспечения планируют запустить и несколько пилотных проектов по внедрению свободного ПО в некоторых регионах России (в данный момент участвовать в эксперименте изъявили желание Пермский край и Татарстан). По итогам использования Linux в школах будут сделаны все необходимые поправки, а к 2010 году правительство РФ планирует оснастить все российские школы свободным программным обеспечением, установив его параллельно с проприетарными образцами. И в таком подходе также есть рациональное зерно: программные продукты компании Microsoft всегда будут использоваться в той или иной степени, поэтому будет совсем неплохо, если российские школьники получат опыт работы не только с Linux-системами, но и с бывшей когда-то своеобразным стандартом Windows.

Правда, Дмитрий Медведев не уточнил, какой свободный программный продукт будет выбран для осуществления пробной миграции. Возможно, он просто еще сам не знает ответа: согласно исследованию Мининформсвязи, из десятка существующих российских вариантов операционных систем семейства GNU/Linux пока ни одна не готова стать полноценной средой для школьного ПО. На данный момент можно выделить три наиболее вероятных кандидата: компания AltLinux, готовящая к осени специализированный школьный дистрибутив, компания ASPLinux, имеющая специальные малобюджетные предложения для образовательных учреждений, и компания Mandriva.ru, являющаяся филиалом международного проекта Mandriva и стремительно ворвавшаяся на российский рынок, выпустив адаптированную к российским реалиям линейку дистрибутивов самого различного назначения.

Скрытая борьба за открытые стандарты

Несмотря на утверждение открытого формата документов Open Document Format (ODF) в качестве международного стандарта ISO в 2006 году, точка в «войне форматов» пока не поставлена. В стане недовольных фактом принятия ODF оказалась корпорация Microsoft, разработавшая собственный аналогичный открытый формат Office Open XML (OOXML), используемый по умолчанию в пакетах офисных приложений Microsoft Office. Ратификация ODF сделала более привлекательными для потребителей альтернативные офисные пакеты, самым популярным из которых является свободный OpenOffice.org (для которого, кстати, ODF является «родным» форматом). Такое положение вещей, конечно, невыгодно Microsoft, для которой MS Office является наиболее весомым источником доходов. Именно поэтому редмондский гигант всеми правдами и неправдами пытается протолкнуть в список международных стандартов и собственный вариант OOXML.

Теоретически нет ничего плохого в том, что появится несколько международных стандартов форматов документов, но дело в том, что «открытость» OOXML вызывает у специалистов очень серьезные вопросы. По различным экспертным оценкам, данный формат соответствует званию открытого международного формата лишь на 60-70%, что может значительно усложнить жизнь пользователей и разработчиков, вынужденных быть привязанными к продукции Microsoft из-за тех самых 30-40% закрытых спецификаций. Впрочем, самого редмондского гиганта данная проблема интересует в последнюю очередь, в связи с чем компанией и проводится титаническая работа по продвижению собственного детища. До конца августа все региональные филиалы Международного комитета по стандартам информационных технологий (INCITS) должны вынести свое решение по поводу OOXML, а на второе сентября намечено голосование в самом INCITS, и результаты региональных решений будут рассматриваться в качестве рекомендательных. Надо сказать, что влиятельные связи Microsoft позволили добиться положительных оценок в некоторых регионах, но недавний предварительный опрос участников INCITS, касающийся США, оказался для OOXML неутешительным. Он показал, что Международный комитет не видит особой необходимости в еще одном стандарте, вызывающем, к тому же, массу вопросов. Впрочем, Microsoft еще может успеть «поработать» с членами INCITS и добиться для OOXML положительного результата.

Linux пришел на рабочий стол

Одним из важнейших факторов, влияющих на широкое распространение GNU/Linux, является наличие в продаже компьютеров с уже установленными свободными системами. В прошлые годы такие предложения были достаточно большой редкостью и часто инициировались не производителями компьютеров, а самими магазинами. Похоже, текущий год можно назвать переломным в данном вопросе: сразу несколько известных брендов выпустили на рынок ноутбуки и настольные ПК с предустановленной операционной системой GNU/Linux. Вслед за компанией Dell, которую к идее выпуска подобных компьютеров подтолкнули сами пользователи, принявшие участие в голосовании на web-сайте компании (Новости, LXF91), активизировались и другие известные игроки рынка. Hewlett-Packard также поспешила объявить о том, что к выпуску в продажу готовится линейка Linux-компьютеров: для корпоративных пользователей будет устанавливаться Novell SUSE Linux Enterprise Desktop 10 SP 1, а для персональных покупателей – Ubuntu-7.04. Несколько дней спустя сообщили о подготовке аналогичных решений и такие компании, как Acer и Lenovo. Первая пока не сообщила о выбранном варианте операционной системы, а вот Lenovo оказалась более разговорчивой: выбор производителя популярных «рабочих лошадок» ThinkPad T-серии пал на SUSE Linux.

Данные факты не могут не радовать, и причина даже не в том, что крупные производители компьютеров наконец признали пригодность Linux к широкому распространению. На самом деле подобные шаги известных брендов являются своеобразным детонатором для цепной реакции: более мелкие производители компьютерной техники не захотят отставать от гигантов индустрии и начнут самостоятельные продажи Linux-компьютеров. Один из первых примеров – недавнее сообщение компании EmperorLinux: они приступили к продажам трех вариантов планшетных компьютеров (Lenovo X60T, Lenovo X61T и Panasonic Tough Book CF-19), «заряженных», по желанию заказчика, операционной системой Windows, Linux или обеими сразу с двойной загрузкой. Причем представители компании гарантируют полноценную поддержку всего аппаратного наполнения планшетов в таких операционных системах, как Fedora, Red Hat, Debian, Ubuntu, SUSE, Slackware и Mandriva. Подобные новости позволяют с оптимизмом смотреть на будущее распространение настольных вариантов Linux среди домашних пользователей и корпоративных клиентов и остается только один вопрос: когда данные модели появятся и в российских магазинах? К сожалению, пока это Linux-изобилие будет доступно покупателям Юго-Восточной Азии, Америки и некоторых стран Европы.

Архивное дело

Наверное, при упоминании SCO Group многие вспомнят жаркие споры по поводу претензий данной компании к Linux-сообществу вообще и к компаниям IBM и Novell в частности. Что ж, похоже, пришло время поставить точку в том далеком споре: федеральный окружной судья штата Юта Дейл Кимболл [Dale Kimball] постановил, что владельцем прав на исходные коды операционной системы UNIX является компания Novell, а не SCO Group Inc. Это значит, что претензии SCO Group к «голубому гиганту» IBM, якобы незаконно использовавшему код UNIX в Linux-системах и тем самым поставившему пользователей Linux в положение обязанных платить SCO, не имеют под собой никаких оснований. Напомню, что изначально еще в 2003 году SCO Group подала в суд на IBM, а через год компания Novell заявила, что IBM тут вообще ни при чем, т.к. права на код UNIX принадлежат именно ей. Судья логично решил, что в первую очередь надо разобраться с делом SCO-Novell, что, в общем, и было сделано в августе текущего года. Представители SCO Group попытались немного сгладить впечатление от своего поражения, заявив, что на самом деле решение суда не касается собственно проблемы, но вряд ли события будут развиваться дальше малоэффективных апелляций. В свою очередь представители Novell сообщили, что другого исхода дела они и не ожидали, поэтому восприняли решение суда довольно сдержанно и без ажиотажа. Также в Novell сказали, что не собираются каким-то образом использовать свою победу в дальнейших спорах по поводу кода UNIX: компания давно вышла из бизнеса, связанного с UNIX, поэтому нет смысла затевать какие-то судебные разбирательства и тратить на них время и деньги. Впрочем, данное заявление послужило успокоением даже не для SCO Group, а, скорее, для других участников рынка: Sun Microsystems использовала код UNIX в своей разработке OpenSolaris и тот факт, что Novell оказалась владельцем прав на него, мог стать отправной точкой для нового судебного дела.

Большая распродажа

Рост популярности свободного ПО подтверждается не только количеством установленных в мире свободных операционных систем, но и повышенным интересом к свободным проектам со стороны коммерческих компаний. Совсем недавно система печати CUPS была приобретена компанией Apple (Новости, LXF95), а август принес еще пару громких покупок. Компания Citrix Systems, Inc. объявила о покупке компании XenSource, Inc., занимающейся разработкой средств виртуализации. Сумма сделки составляет около $500 млн., но это не самое интересное: Citrix является одним из наиболее крупных партнеров Microsoft, поэтому приобретение XenSource, свободные разработки которой входят во многие Linux-дистрибутивы и даже включены в состав ядра Linux, заставляет задуматься о дальнейшем будущем Xen в качестве одного из самых популярных средств виртуализации в Linux-системах. Еще одна августовская покупка была сделана компанией Sourcefire, Inc., которая приобрела за неназванную сумму проект ClamAV, являющийся наиболее популярным вариантом свободного антивирусного ПО, также по умолчанию входящего практически во все солидные дистрибутивы Linux. Правда, участники этой сделки сразу предупредили, что ClamAV по-прежнему будет развиваться под свободной лицензией GPL.

GPL на перспективу

Как и положено в нашем стремительном мире, все течет, все меняется. В том числе – и способы работы с программным обеспечением. Если раньше все приложения устанавливались на локальный компьютер, то сегодня сетевой мир все больше переходит на рельсы Web 2.0, когда приложение выполняется на удаленном сервере, передавая пользователю лишь результаты своей работы. Традиционно среди подобных проектов довольно много развивающихся по принципам свободного ПО, но интернет-специфика некоторых свойств этих программ не позволяет свободной лицензии GPL охватить все аспекты разработки и использования данных приложений. Именно для таких случаев существует GNU Affero GPL (AGPL), регулирующая порядок работы с сетевым свободным ПО. В середине августа был опубликован второй черновик лицензии AGPLv3, в основу которой лег текст вышедшей чуть ранее GNU GPLv3. Основным отличием от прототипа стали пункты, касающиеся изменения исходных кодов свободных web-приложений. Согласно AGPL, разработчик, взявший за основу какой-нибудь свободный проект и создавший на его базе собственный онлайновый сервис, обязан опубликовать исходные тексты своих дополнений или модифицированных компонентов также под свободной лицензией. Пока лицензия AGPL не очень известна широкой публике, но учитывая все увеличивающуюся долю сетевого ПО можно предположить, что в ближайшем будущем она будет не менее популярна, чем собственно GPL.

Новости короткой строкой

» Фонд TheLinuxFund.org объявил о выпуске в обращение кредитных карт Visa, с помощью которых будет осуществляться финансовая поддержка свободных проектов.

» Компании Acronis и Novell приступили к совместной разработке системы восстановления и резервного копирования данных для операционных систем SUSE Linux.

» Компания Microsoft подала заявку на признание своих лицензий Shared Source, под которыми распространяются некоторые программы редмондского гиганта, как открытых.

» Релиз операционной системы Red Hat Global Desktop Linux отложен на неопределенный срок в целях улучшения мультимедийных функций дистрибутива.

» Компания Hewlett-Packard объявила об открытии кода Parallel Compositing API, предназначенного для создания средств управления большими объемами данных.

» Компания Google вошла в состав Open Invention Network, став одним из наиболее влиятельных защитников свободного сообщества от патентных угроз, а Nokia Siemens Networks объявила о входе в состав Linux Foundation – из желания принять активное участие в разработке единой мобильной Linux-платформы.

» Компания Motorola сообщила, что через два года около 60% производимых ею телефонных трубок будут использовать ПО на базе Linux.

» 15 августа этого года популярному проекту GNOME исполнилось 10 лет.

IBM

МОЖНО ЛИ ДОСТИЧЬ НОВЫХ ВЫСОТ БЕЗ ОПОРЫ НА СЕРВИС-ОРИЕНТИРОВАННУЮ АРХИТЕКТУРУ

как ключевую составляющую вашей ИТ-инфраструктуры? Используйте IBM WebSphere для объединения и интеграции существующих у вас приложений, информации и аппаратных систем. IBM WebSphere позволяет выделить элементы бизнес-процессов и ИТ-инфраструктуры, преобразовав их в гибкие компоненты, которые можно многократно использовать при изменении условий ведения бизнеса. «Аэрофлот» при помощи IBM внедрил решение по интеграции разнообразных критически важных для бизнеса информационных систем, что позволило авиакомпании более оперативно реагировать на требования российского рынка. Самым важным в этом примере является способность реагировать на изменения, используя уже имеющиеся в наличии средства.

Реклама

WebSphere

Подробнее на: **ibm.com**/software/ru/campaign/websphere

IBM, логотипы IBM и WebSphere являются товарными марками или зарегистрированными товарными знаками International Business Machines Corporation в США и/или других странах. © 2007 IBM Corporation. Все права защищены.

LXF Репортаж

Итоги мастер-классов: Сибирь и Урал

Мастер-классы по Linux-технологиям, проводимые компаниями LinuxCenter и Mandriva.ru при поддержке журнала Linux Format, продолжают свое плодотворное движение по просторам России. Очередной этап прошел в Сибири и на Урале и следует сказать, что практические результаты данных мероприятий оказались довольно ощутимыми. Живой интерес к свободным операционным системам проявили почти во всех регионах, причем не только представители IT-отрасли и бизнесмены, но и, что еще более важно, чиновники региональных органов власти. О мастер-классах, проходивших в Новосибирске, мы уже писали (LXF95), поэтому сразу перейдем к мероприятиям, прошедшим в Омске.

Отличительной чертой мастер-классов в этом городе стало повышенное внимание к Linux-технологиям у представителей областного правительства. Помимо того, что для проведения мероприятия чиновники «пожертвовали» собственным залом заседаний и предоставили компьютеры, позволив установить на них операционную систему Mandriva, представители омских властей принимали активное участие и в самих мастер-классах, живо интересуясь вопросами технического и правового характера, неизбежно возникающими при серьезном разговоре о свободном ПО. В роли докладчика по уже сложившейся традиции выступил директор компании LinuxCenter Павел Фролов. Вообще, если говорить об общем впечатлении, то пример омских областных властей показал, что чиновники вполне готовы к широкому внедрению свободного ПО в государственных учреждениях и вопрос лишь за малым – нужен готовый программный продукт, способный без особых проблем заменить использовавшиеся ранее проприетарные дорогостоящие операционные системы и программные средства. Поэтому исключительно важное значение имел тот факт, что в процессе проведения мастер-класса все присутствовавшие могли самостоятельно попробовать поработать в операционной системе Mandriva, предварительно установленной на каждый компьютер зала заседаний, и оценить, насколько простым и доступным стало свободное ПО в последние годы. Кстати, о легкости миграции на Linux косвенно говорит и тот факт, что переоснащение всех имеющихся в предоставленном помещении компьютеров на использование операционной системы Mandriva было осуществлено всего за один день без каких-либо проблем силами местных системных администраторов и самого докладчика. В заключении рассказа об омских мероприятиях можно отметить, что разговор о свободном ПО вышел далеко за границы мастер-классов: в последующие дни уже в рамках Сибирского Интернет Форума Павлу Фролову пришлось рассказывать о Linux в

Компания «АСП», г. Екатеринбург

КОМПАНИЯ АСП® 1С:Франчайзинг

Компания «АСП» входит в состав Группы компаний «АСП» и работает на ИТ-рынке Уральского Федерального округа с 1992 года.

Компания «АСП» специализируется на автоматизации бизнеса и предлагает следующие услуги:

» **Продажа и внедрение программ «1С» для автоматизации учета и управления предприятием.**
Компания «АСП» готова не только продать вам программное обеспечение системы «1С», но и помочь в выборе, освоении, адаптации, при необходимости – разработать решение, отвечающее именно вашим потребностям. К вашим услугам высококвалифицированные сертифицированные специалисты, которые настроят программы и дадут рекомендации по постановке бухгалтерского, складского и внутреннего учета на вашей фирме.

» **Продажа системного программного обеспечения.**
Компания «АСП» – один из ведущих поставщиков лицензионного программного обеспечения операционных систем, офисных приложений, графических пакетов, утилит, средств разработки, программных продуктов для научно-технических расчетов и моделирования.

Мы предлагаем продукты иностранных и российских разработчиков – Adobe, Autodesk, Borland, Citrix, ComputerAssociates, Corel, Crystal Decisions, Intel, Macromedia, MathSoft, Mathworks, Microsoft, Novell, Pervasive Software, Symantec, Veritas, ABBYY, Promt, Диалог Наука, Лаборатория Касперского и пр.

» **Комплексная автоматизация предприятий розничной торговли.**
Компания «АСП» является официальным партнером компании «АСТОР ВЦ» (г. Москва), специализирующейся на разработке программных продуктов для автоматизации торговли.

Индивидуальное, групповое и корпоративное обучение на базе авторизованного Учебного центра «1С»; бизнес-консалтинг.

Наша главная задача – решать проблемы заказчика качественно, с минимальными временными и финансовыми затратами.

Компания «АСП» сертифицирована по стандарту системы менеджмента качества ISO 9001:2000.

г. Екатеринбург, ул. Белинского, 163в, 3-й этаж
Тел.: (343) 220-20-20
www.asp-1c.ru

Что будет LINUXLAND/SOFTOOL/SITOP—2007 2–5 октября, Москва, ВВЦ, павильон 69

Выставка LinuxLand – единственное в России мероприятие, в экспозиции которого представлены ведущие российские и международные компании, предлагающие решения и услуги на базе технологий с открытым исходным кодом. Из года в год LinuxLand объединяет таких поставщиков ПО Open Source и сопутствующих услуг, как Mandriva, IBM, Novell, R-Style, HP, Oracle, ASPLinux, Linux-Online (разработчик Linux XP), НПО «Сеть» (разработчик MOPSLinux), Bitrix, ПРОМТ, Etersoft, LinuxCenter.ru, журнал Linux Format, образовательный центр Unix Education Center и другие.

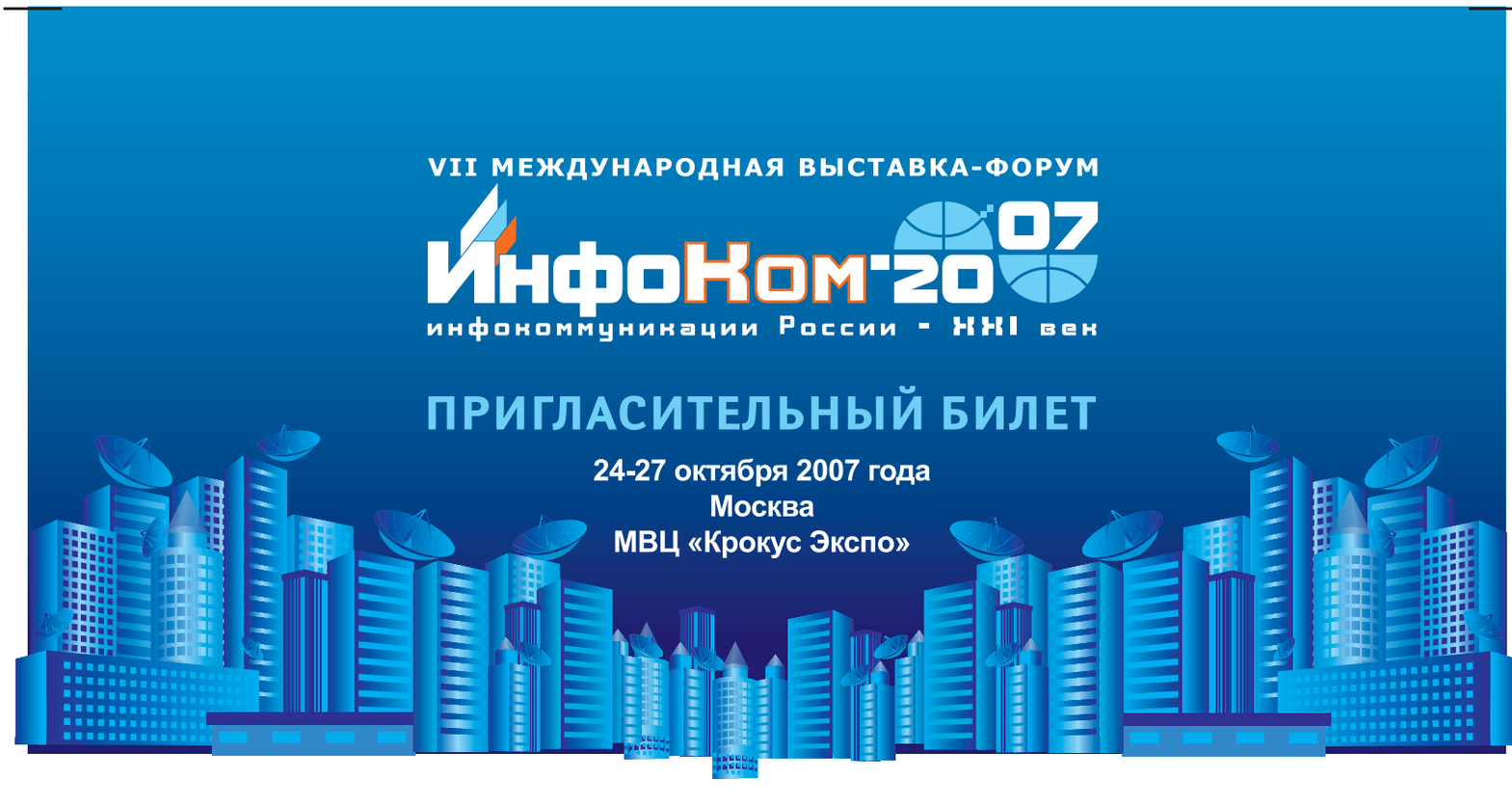
В этом году в рамках деловой программы выставок SofTool/LinuxLand состоится I Международная конференция «Стандартизация информационных технологий и интероперабельность. SITOP–2007», включающая секцию «Стандарты открытых систем и ПО с открытым кодом как инструмент свободной реализации ИТ-инфраструктур». Конференция отражает важнейшие тенденции в области развития ИТ, практики использования программных и аппаратных продуктах в различных сферах экономики и государственного регулирования сферы информационных технологий.

Омском Союзе Предпринимателей, на радио «Эхо-Москвы Омск» и на нескольких местных телеканалах. Не менее впечатляющими оказались и результаты: по итогам мастер-классов и последующих мероприятий в Новосибирске и Омске были открыты центры компетенции Mandriva в НГУ, НГТУ, ИЯФ, при правительстве Омской Области и в Омском государственном техническом университете.

Уральский этап мастер-классов прошел в Екатеринбурге в офисе компании АСП (www.asp-1c.ru). Количество желающих принять участие в этом событии значительно превысило ожидаемое, в связи с чем организаторам пришлось назначить дополнительные занятия, тем самым расширив двухдневную программу до трех дней. Основные темы мастер-классов касались всех наиболее значимых сегодня вопросов: базовые навыки работы с Linux-системами на примере дистрибутива Mandriva, использование свободного ПО в образовательной сфере и возможности Linux в корпоративном секторе. Подобная широта охвата тем сказалась и на составе участников: в аудитории присутствовали представители муниципалитета и образовательных учреждений, работники IT-подразделений крупных компаний и просто пользователи операционных систем GNU/Linux, проживающие не только в самом Екатеринбурге, но и в городах свердловской, пермской и челябинской областей. Следует также отметить, что на этот раз Павлу Фролову не пришлось выступать в одиночестве: на мастер-классах смогли рассказать о своем опыте внедрения Linux-систем местные системные администраторы и преподаватели школ. Если говорить об итогах екатеринбургских мастер-классов, то они также довольно ощутимы: помимо открытия в городе официального представительства LinuxCenter-Урал, взявшего на себя поставки всей линейки продукции компании Mandriva.ru (и не только), в Екатеринбурге появились и центры компетенции Mandriva, организованные на базе компании CIFT (www.cift.ru) и при муниципалитете Екатеринбурга.

Подытоживая все вышесказанное, можно отметить, что данные мастер-классы оказались весьма своевременны для Сибири и Урала, а зачастую Linux-зерна падали уже на хорошо подготовленную почву: внедрение свободного ПО на местах уже началось силами самих пользователей и централизованная помощь со стороны LinuxCenter и Mandriva.ru может оказаться тем связующим звеном, которое позволит поднять процесс внедрения Linux на качественно новый уровень. LXF

» При подготовке данного выпуска новостей были использованы материалы сайтов: Приоритетные национальные проекты, InfoWorld, vnunet.com, ars technica, Free Software Foundation, Citrix и других, а также наших собственных корреспондентов.



КОРПОРАТИВНЫЕ СЕРВЕРЫ
И СИСТЕМЫ ХРАНЕНИЯ ДАННЫХ

(812) 327-5960
(495) 232-9230
info@trinitygroup.ru

TRINITY
CORPORATE IT PROJECTS

- Анализ существующей IT инфраструктуры
- Разработка технического задания
- Проектирование, монтаж, внедрение
- Комплексное управление IT инфраструктурой
- Катастрофоустойчивые решения

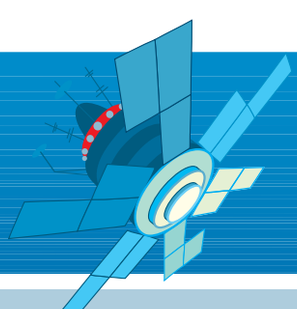
Серверы

под Linux
FreeBSD
Solaris x86

для баз данных, интернет шлюзов, WEB-приложений, кластеры для научных расчетов

Мы делаем бизнес успешным

www.trinitygroup.ru
Информационные технологии от экспертов



организатор:
FORMIKA

При поддержке Министерства информационных технологий и связи Российской Федерации

премьер партнер:
СВЯЗЬ ИНВЕСТ

официальные спонсоры:
РОСТЕЛЕКОМ
МТТ

спонсоры:
NW GROUP
ЭЛИКС
ARD

регион-партнер:
Краснодарский край

информационные партнеры:
Сети и системы связи
computerworld
TECHNOPOLIS
ФОТОН-ЭКСПРЕСС
РУССКАЯ РЕДАКЦИЯ
COMNEWS
ТВ
cnews
Связьинвест
ПЛАС
Издания Максимова
Информационный мост
T-Comm
Техника Связи
ИНФОСФЕРА
Технологии и средства связи
Computer Bild поможет.
ИКС
Горячая линия-Телеком
Сети
softline
ВЕК качества
СВЯЗЬИНФОРМ
PCWEEK
МОБИЛЬНЫЕ ТЕЛЕКОММУНИКАЦИИ
ТЕЛЕспутник
Издательский дом connect
IT-бизнес
CRN
Журнал электросвязь
Журнал сетевых решений LAN
Мобильные системы

LINUX FORMAT

Обзоры

Новинки программного и аппаратного обеспечения в описании наших экспертов

Алексей Федорчук
Свою первую (и последнюю) программу написал еще на Алголе

SCO'тский вопрос

В последние дни в новостях и на форумах интенсивно обсуждается судьбоносное решение суда по делу между Novell и SCO, а также комментарии заинтересованных сторон, последовавшие за ним. Повсеместно задаются вопросы: хорошо это или плохо? А я задам другой вопрос: а есть ли предмет для обсуждения?

Действительно, а что нового мы узнали из решений и комментариев? Что права на товарный знак UNIXTM (а это слово – не более, чем товарный знак) принадлежат консорциуму The Open Group, вроде бы никто никогда не сомневался. А уж что такое исходный код UNIX – тайна сия велика есть. Особенно с учетом того, что последним чистым Unix'ом от AT&T был System V R3 – потому что в R4 было столько BSD-кода, что на ум неизбежно приходил вопрос – где кончается полиция и где начинается Беня Крик? На который резонные люди из Одессы дали ответ много лет назад...

Так что, дорогие коллеги, мы просто являемся зрителями очередного матча по национальному американскому виду спорта – публичному судилищу. Кто судит, кого, за что и почему – никого не волнует. Да и между кем матч проходит – тоже по большому счету не важно. Главное – нагнетать азарт зрителей и болельщиков. Как бы для разминки...

И зададимся последним вопросом: а нам оно нужно? Не лучше ли размяться на перекладине, помолотивши кулаками по макиваре, помахавши бокэном... (нужное дописать)? А потом со свежими силами взяться за свое непосредственное дело. И пусть нас не волнуют ихних глупостей...

alv@posix.ru

Сегодня мы рассматриваем...

22 Defcon
Все когда-нибудь умирают. Под впечатлением от популярной в восьмидесятых War Games, индийская компания Introversion решила разрабатывать стратегию взаимоуничтожения, а не старые добрые шахматы.

24 Fedora 7
Со старейшим дистрибутивом происходит много интересных изменений: хорошо это или плохо?

25 GP2X
Нам пришлось потрудиться, но в конце концов мы заполучили открытую игровую консоль с питанием от батареек и сумели написать по ней этот обзор.

27 Nero Linux 3
Nero Burning Rom – стандарт для Windows, а это – третье поколение средств записи CD/DVD в Linux. Проблема в том, что Linux и так не испытывает недостатка в этих самых средствах.

28 TwonkyMedia 4.4
Любой пылящийся Linux-компьютер можно быстро превратить в клиент потокового вещания с помощью TwonkyMedia. Да, PSP, которой вы давно не пользуетесь, тоже подойдет.

GP2X с. 28

› **Ух ты, прямо как из 2007 года! Нам в руки попалась игровая консоль на базе Linux.**

Nero с. 28

› **Третье поколение Nero Burning Rom – то ли это, что нужно Linux?**

НАШ ВЕРДИКТ: пояснение

Все попавшие в обзор продукты оцениваются по одиннадцатибальной шкале (10 – высшая оценка, 0 – низшая). Как правило, мы оцениваем функциональность, производительность, простоту использования и цену, а для бесплатных программ учитывается документация. Кроме того, мы всегда выставляем общую оценку, демонстрирующую наше отношение к продукту.

Выдающиеся решения могут получить престижную награду «Top Stuff». Номинантами становятся лучшие из лучших – просто высокой оценки здесь недостаточно.

Рассматривая свободное ПО, мы обычно указываем предпочтительный дистрибутив. Иногда это означает компиляцию из исходных текстов, но, если разработчики рекомендуют Autopackage, мы следуем этому совету.

LINUX FORMAT Вердикт

Google Earth
Разработчик: Google
Сайт: http://earth.google.com
Цена: Бесплатно по закрытой лицензии

Функциональность	10/10
Производительность	9/10
Простота использования	9/10
Документация	9/10

» *Если весь мир – сцена, то Google Earth – театр. Простая в использовании, захватывающая и ободряюще практичная программа.*

Рейтинг 9/10

Defcon

Погибли все. В смысле, кроме **Пола Хадсона** с его любимым 13-ногим тараканом: должен же был кто-то написать этот обзор…

Вкратце...

» Игра-стратегия ядерной войны за мировое господство. См. также: *Darwinia* и *Uplink*.

Пока я пишу эту статью, Тони Блэр как раз передает высшую исполнительную власть Гордону Брауну, и технически это означает, что мы примерно на полчаса можем остаться вообще без премьер-министра. А вдруг в этот момент кто-нибудь возьмет да предпримет полномасштабную ядерную агрессию на всю Западную Европу?

Defcon – новейшая версия игры от индийской компании Introversion – помещает игрока в правительственный бункер, где пальцы сами тянутся к красной кнопке. У каждого участника есть ракетные шахты, пригодные как для нападения, так и для защиты; радарные базы, ощупывающие небо; аэродромы с готовыми к взлету истребителями; плюс подводные лодки, авианосцы и военные корабли для борьбы за морское господство.

› **Чтобы причинить максимальный урон, нацельте ракеты на разные города.**

Defcon 5

Международные договоры не позволяют бомбить врага прямо сейчас, поэтому у нас в запасе около десяти минут на организацию своей обороны и развертывание флотов. Комбинация военных кораблей и авианосцев оптимальна для защиты, но ослабляет наступательную мощь. Можно успеть договориться с агрессорами – перемирие дает шанс перегруппировать свои силы на всей планете, а удачная дипломатия еще и способна уменьшить число потенциальных угроз. Естественно, полностью доверять союзникам нельзя, поэтому опытный военачальник всегда будет держать возле дружественных берегов несколько подлодок… так, на всякий случай.

> «После пятого попадания от Мехико осталось лишь зеленое облако».

В этот период нам не видны силы противника, но путем умозаключений можно предположить наиболее вероятные направления ударов. Время истекает стремительно, и если мы не сумеем правильно расставить силы до наступления *Defcon* 3, другого шанса у нас не будет.

› **Для контроля за морями надо объединить силы авианосцев и военных кораблей, а субмарины тайком подвести поближе к берегам, чтобы нанести удар наверняка.**

Defcon 4

Все радарные установки активируются, показывая, как на ладони, детальные виды вражеской территории. Мои войска через границу с Афганистаном внимательно изучают Россию, где бункеры Грэма «Degville» Моррисона расположены по периметру укрепленной Москвы. Остальная часть страны осталась незащищенной; гражданам остается вручную рыть атомные бомбоубежища и гадать, спасут ли их от неизбежных радиоактивных осадков жалкие пакеты из оберточной бумаги.

Defcon 3

К востоку от Токио, в непосредственной близости от одного из моих флотов, Грэм развернул две флотилии своих авианосцев. Кажется, война не заставит себя долго ждать. Пока я потягиваю шампанское в правительственном бункере-люкс под Тегераном, орудия наших кораблей открывают огонь. Щелчком тумблера я поднимаю на крыло десять эскадрилий бомбардировщиков с ближайшего аэродрома для воздушной поддержки тихоокеанского флота. Ник – в своей далекой Южной Америке – угроза пока неблизкая.

Defcon 2

От флотов Грэма остались чадящие обломки, это славная победа Народно-Демократической Республики Хадзилла. Но пока мои граждане праздновали, с тихоокеанского авианосца пришло сообщение о том, что подлодки Ника приближаются, и я отдаю приказ истребителям обшарить океан. В это время на главной арене боевых действий Грэм перевел все свои шахтные комплексы в состояние противовоздушной обороны: вероятно, надеется защитить себя от объединенной атаки моих бомбардировщиков и ракет. Как он заблуждается…

Defcon 1

Оружие готово, заряды активированы, и красная кнопка нажата несколько раз, для вернос-

Только для пацифистов

Defcon нетрудно перенастроить, и тот, кому претят смерть и разрушения, может поиграть в режиме Christmas, рассылая вместо бомб рождественские подарки. Отправляйте запряженные оленями санки людям, которым вы желаете добра, а по их прибытии на место будете получать сообщения типа: «Сан-Франциско: осчастливлено 3,5 миллиона ребятишек». Гениально.

Зловещее звучание тишины

Defcon выполнена в традициях Introversio: игра полностью двухмерная, карта мира линейная со слабой подсветкой материков по контуру. Яркие белые грибовидные облака ядерных разрывов вскоре сменяются зловещими зелеными пятнами радиоактивного заражения, и с первого взгляда можно определить зону наибольшего поражения. Но вот от чего волосы встают дыбом, так это от звукового сопровождения: хоральные песнопения перемежаются отдаленными криками и стонами умирающих, при разрыве бомбы слышен глухой рокот – словно и вправду что-то взорвалось поблизости. Виртуальные сограждане погибают, но Вождь холоден и отстранен, он должен мстить врагам и спасать оставшихся в живых. Нелегкая, прямо скажем, задача – сохранять хладнокровие, наблюдая агонию мира!

› ***Defcon* в разгаре: десятки ракет воздухе, миллионы людей гибнут, но – то ли еще будет!**

ти. И тут – катастрофа: все восемь выпущенных мной ракет сбиты прямо в небе коварными системами обороны Грэма. Итак, оставшись между Грэмом с одной стороны и Ником с другой, я поступаю так, как поступил бы любой великий полководец: заключаю союз с врагом.

Война!

Соединенные Штаты Дегзиллы занимают всю Россию и Азию, от Москвы до Токио. Щупальца ЗлоНика протянулись через всю Южную Америку, его штаб-квартира находится поблизости от Мехико, на другой стороне планеты. Пока наши войска находились вне зоны дальности ракетного удара, Ник, похоже, воспользовался нашей с Грэмом мини-перестрелкой и перегруппировал свои силы – две флотилии его подлодок всплыли: одна под Бомбеем, другая – у берегов Норвегии. И вот они – за какие-то 20 секунд – изготовились к атаке, нацелив свои ракеты на наши города.

Одновременно с открытием огня московскими батареями, завязалась еще одна тихоокеанская битва – на этот раз между моими силами и флотом ЗлоНика. Снова самолеты, снова тонущие корабли, снова крики, и вскоре Тихоокеанский флот Ника отправился в океанскую пучину на встречу с разбитыми кораблями Грэма. Пока я бился с Ником, подлодки Грэма всплыли в районе Мехико и дали залп. Безуспешно: защитные комплексы Ника сбили все ракеты еще на подлете. К несчастью, свою территорию Грэм защитить не смог: удар по Ленинграду унес 11,2 миллиона жизней.

ЗлоНик умудрился наскрести несколько штурмовиков в Колумбии, и они потопили беззащитные лодки Грэма одну за другой. В следующие несколько секунд погибли миллионы во всем мире. Поражен Киев с 5,6 миллионами несчастных. Пилоты Грэма пытались атаковать флот Ника у берегов северной Сибири, но были сбиты, даже не долетев до цели.

Москва в руинах: 12,3 миллиона трупов.

Бомбей разбит: 9,1 миллиона погибших.

Токио больше нет: 15,9 миллиона жертв.

Возмездие

Пока ЗлоНик превращал восточную Россию в радиоактивную пустыню, мои подлодки наконец достигли Мехико. В тот же миг из моих шахт вырвались ракеты, бомбардировщики взмыли в небо, и карта мира покрылась пятнами разрывов – и все вокруг Южной Америки. Минуты шли, я прихлебывал чаек да болтал в чате с Джереми 'Dr Strangelove' Фордом [Jeremy 'Dr Strangelove' Ford] из *PC Format*. И только когда мои ракеты приблизились к берегам Флориды, лодки дали свой залп: эта синхронная атака была задумана с таким расчетом, чтобы никакая оборона Ника не смогла справиться со страшным ударом.

Сработало: в Мехико убиты 15 миллионов. Минутой позже та же участь постигла тех, кто праздновал спасение. После пятого попадания вместо Мехико на моей карте осталось лишь зеленое облако радиоактивности.

Измена

Мои отцы-командиры уже попивали праздничное шампанское через отверстия в своих антирадиационных костюмах, когда в бункере зазвонил телефон, да не простой – особый, красный, который я поставил в день заключения союза с Грэмом. Недоброе известие: Соединенные Штаты Дегзиллы распались надвое, и мой бывший союзник снова нацелил ракеты на мои города, желая вписать себя – и только себя – в анналы Истории.

Конец

Пишу эти строки в своем атомном бункере. Наш главный академик считает, что на поверхность можно будет подняться только к Новому году. В смысле, Новому 2077 году. Империя Хадзилла выиграла войну. Да, три четверти планеты теперь непригодны для жизни, но ведь победа за мной! Да здравствуют Великий Вождь и его любимый таракан Тимми! Сроду не воевал с таким азартом… LXF

Грэм считает...

«Пол как-то позабыл упомянуть, что все остальные битвы выиграли мы с Ником. Ну, то есть, одну или две. За 10 старых колониальных фунтов *Defcon* – просто блеск!»

Зарядов не жалеть

Defcon – удивительно скоротечная игра. Суматоха Defcon 1 отмечена морскими баталиями, рысканьем истребителей и безумными альянсами, но это ничто по сравнению с хаосом финальной фазы игры. Ракеты – порой до 30 разом – разлетаются во всех направлениях, и нужно тщательно соблюдать баланс между числом установок ПРО и наступательными силами. Засев в глухую оборону, можно остаться вообще без ракет, ведь каждая шахта, получив три попадания, уходит в небытие.

LINUX FORMAT Вердикт

Defcon

Разработчик: Introversion
Сайт: www.everybody-dies.com
Цена: $19,50

Сюжет	8/10
Графика	9/10
Увлекательность	7/10
Цена	8/10

» *Все погибли. Никто не выжил. Но от слез до смеха всего один шаг…*

Рейтинг 8/10

Fedora 7

Шесть месяцев – достаточно долгий срок для стремительно меняющейся Линуксландии. **Энди Хадсон** выясняет, насколько он долог для Fedora.

Вкратце...

» Популярный Linux-дистрибутив, теперь с Live CD. См. также: Ubuntu, OpenSUSE и Mandriva.

Вот и настало снова то время года, когда очередная версия Fedora вступает в борьбу за пользователя. Fedora Core 6 была популярна, и не столько из-за бравого прозвища Zod, сколько из-за демонстрации широкого шага Fedora. Обещали, что Fedora 7 шагнет еще шире. Сбылось ли обещание?

Как всегда в версиях Fedora, включено основное ПО сторонних проектов, обновленное за время цикла разработки Fedora 7. Здесь и Gnome 2.18, и *OpenOffice.org 2.2*, *Firefox 2.0.0.4*, ядро 2.6.21 и иже с ними. Можно долго мусолить тему, насколько удобнее новая версия для пользователя настольного ПК, да и для любого пользователя, но тогда мы рискуем упустить главные достоинства этой версии.

Во-первых, и это главное, хочется отметить новые разновидности Fedora. Первая из них – старый добрый DVD, 3 ГБ которого заняты разнообразным ПО. Выпущены и два Live CD, оба официальные: для Gnome и для KDE. Что особенно радует – *GParted* присутствует на обоих, а это значит, что проблемы NTFS-разделов больше нет.

› ***Revisor*** **– нововведение, позволяющее самостоятельно создавать Live CD или инсталляционные диски с полным контролем над процессом.**

> «Усилия Fedora по включению KVM сильно укрепили их позиции.»

На обоих Live CD ставшая уже привычной функция прямой инсталляции на жесткий диск. Безупречно встроенная в них *Anaconda* – простой и удобный для пользователя инсталлятор. Единственное неудобство – набор пакетов ограничивается теми, что присутствуют на Live CD. Но мы уверены, что со временем этот недостаток будет исправлен. Приятно отметить прогресс Fedora в этой области.

Вероятно, самые заметные нововведения в Fedora 7 – это *livecd-tools*, инструментарий для сборки своей собственной редакции дистрибутива, и *Revisor*. С помощью *Revisor* можно создать как Live CD, так и полноценный набор инсталляционных дисков. Интересно, что перекроить по своему вкусу можно любую редакцию Fedora.

Более того, *Revisor* позволяет полностью контролировать процесс создания дистрибутива, вплоть до выбора пакетов. Никогда еще собрать собственную версию Fedora не было так просто.

Пол считает...

«Это, конечно, не прорыв, но тех, кто западает на внешний блеск, порадуют новые темы. Понадеемся, что разработчики будут целить выше в следующий раз.»

OK, KVM

Интересно взглянуть на интеграцию *KVM* в Fedora 7. В обзоре Red Hat Enterprise Linux 5 (LXF92) мы похвалили *virt-manager*, инструмент, предназначенный в Red Hat для управления виртуальными машинами *Xen*. А теперь *virt-manager* значительно усовершенствован, и может работать с машинами *KVM*. Мы едва не рискнули предположить, что Red Hat/Fedora одинаково относятся и к *Xen*, и к *KVM*, но склонились к тому, что *KVM* станет доминирующей платформой в Fedora/RHEL – хотя бы из-за более плотной интеграции на уровне ядра. Усилия Fedora по включению *KVM* сильно укрепили их позиции, и мы с нетерпением ожидаем новых успехов.

В завершение обзора необходимо упомянуть о том, что больше не существует ни Fedora Core, ни Fedora Extras – просто Fedora. Это свидетельствует о движении Red Hat по направлению к полному открытию для сообщества – в последнее время компанию часто обвиняли в отходе от общих принципов, но теперь Fedora использует отдельную инфраструктуру сборки вне Red Hat, и весь исходный код содержится в независимом хранилище; короче, любой желающий теперь может приложить руку к коду Fedora.

Fedora 7 – важная веха в развитии проекта Red Hat/Fedora, и, на беду для Ubuntu, состязание на этом пространстве вновь становится захватывающим. LXF

LINUX FORMAT Вердикт

Fedora 7

Разработчик: The Fedora Project
Сайт: http://fedoraproject.org
Цена: Бесплатно под GPL

Функциональность	10/10
Производительность	9/10
Простота использования	9/10
Документация	8/10

» *Стоит взглянуть. Тому, кто отвернулся было от Fedora, самое время поворачиваться обратно.*

Рейтинг 9/10

GP2X

Linux на игровой консоли? Думаете, мы шутим? **Майк Сондерс** присматривается к крошечному самопальному наладоннику с Туксом внутри.

Вкратце...

» Карманная игровая система, рассчитанная на самостоятельное изготовление компьютерных игр и эмуляцию.
См. также: Nintendo DS и Sony PSP.

Грэм считает...

«Такое ощущение, что GP2X – всего лишь рабочий прототип. Открытая платформа – это великая идея, да и аппаратная часть обладает потенциалом, но все испортил негодный дизайн. Может быть, следующее поколение что-нибудь исправит.»

Поднимите руки те, кто помнит Indrema – разрекламированную и быстро забытую попытку сделать игровую консоль с Linux. Ее неудача не добавила Linux популярности как игровой платформе, но вот спустя шесть лет корейская компания GamePark Holdings делает вторую попытку, с GP2X. Этот маленький легкий наладонник, оснащенный двухъядерным процессором на 200 МГц и 64 MB ОЗУ, призван доказать, что разработка игр может быть и хобби. GP2X работает на Linux и позволяет писать или портировать игры, не заботясь о лицензиях и гарантиях.

Примерно такого же размера, что и Nintendo DS Lite, GP2X уютно располагается в руках, органы управления легко доступны. К несчастью, D-пульту дизайнеры предпочли ужасный псевдоаналоговый джойстик. Это выступающий кружок, который вертится (бестолково) под большим пальцем и с пугающей легкостью сбивается не в ту сторону. GamePark включила в комплект замену – колпачок на джойстик, который после часовой игры намертво впечатывает аббревиатуру «GP2X» в подушечку большого пальца. Управлять можно и в привычном SNES-стиле, кнопками L и R.

Корпус устройства местами слабоват: например, чтобы при перемещениях консоли не отключалось питание, мы были вынуждены поместить в батарейный отсек кусочек пластика. Порты расширений (наушники, USB, питание) прикрыты тоненькими клапанами, подвешенными на хлипких резиновых ниточках. Непохоже, что эта конструкция продержится долго. Дисплей GP2X 320x240 с подсветкой очень четкий, но при разряженных батареях резко портится. Определенно, для этого устройства необходим аккумулятор: в среднем батареи садятся за жалкие четыре часа игры.

› **Этот так называемый «джойстик» изрядно портит картину, правда?**

› **Набор функций солидный, но ни в одной из них GP2X не преуспела.**

Но это аппаратная сторона GP2X; а что она умеет делать? Для разработки ПО имеется инструментарий GNU вместе с *SDL* (дополнительные библиотеки, например, *Allegro*, можно добыть у сторонних разработчиков). Если у вас есть *SDL*-игра, портировать ее, скорее всего, не составит большого труда, если позаботиться об ограниченных по сравнению с полноценным компьютером органах управления и учесть малый размер экрана у консоли. В комплекте с GP2X идут игры для демонстрации возможностей портирования, например, веселая и легкая *SuperTux*. Есть демо-вариант *Payback* – впечатляющего 3D-клона GTA – и другие развлекательные штучки.

Поиграл – посмотри

Еще один элемент маркетинга GP2X – воспроизведение медиа. Вставляешь SD-карту – и смотришь фильмы в форматах AVI, DivX и Xvid. Файлы фильмов открываются, встроенные динамики выдают неплохой звук, но просмотр не особо комфортный: видео слишком часто «спотыкается». Воспроизведение MP3, напротив, замечательное, хотя изображение с экрана в промежутках между дорожками исчезает. Персональный медиаплеер из GP2X неважный, к тому же многие фильмы необходимо конвертировать для приведения к меньшим размерам и битрейту. А 19-секундная загрузка немало раздражает, если всего-то хочется послушать пару песен.

Но главный козырь GP2X – это эмуляция. На http://archive.gp2x.de можно найти сотни программ для этого устройства, включая многие от классических консолей. Мы добились отличных результатов с эмуляцией Game Boy, Game Gear и Mega Drive – ничего нет круче, чем *Streets of Rage II* прямо на ладони. SNES-эмуляция гораздо хуже, с глюками, но разработчики сулят GP2X исправления и светлое будущее.

GamePark неплохо поработала, так много втиснув в относительно дешевое устройство, хотя аппаратная часть оставляет желать лучшего. Убрать дурацкий джойстик и поставить продуманный D-пульт, добавить аккумулятор и взять пластик пожестче – выйдет чудесная консоль. Пока же это только забава для технарей. Фанаты ретро могут исследовать арену эмуляции и посмотреть, запускаются ли их любимые игры. Ну, а тот, кто ищет персональный аудиоплеер или полноценную игровую машину, может без сожаления отвернуться – ничего подобного DS или PSP здесь нет. LXF

LINUX FORMAT Вердикт

GP2X

Разработчик: Game Park
Сайт: www.gp2x.com
Цена: $189.99

Функциональность	5/10
Производительность	6/10
Простота использования	6/10
Цена	8/10

» *Мечта любителя ретро-игр, если забыть о качестве изготовления и отсутствии аккумулятора.*

Рейтинг 6/10

Nero Linux 3

Linux-версии популярных программ – это замечательно. **Грэм Моррисон** проверяет, созрел ли прожиг компакт-дисков для промышленной эксплуатации.

LINUX FORMAT на DVD
» Nero Linux 3 Демо

Вкратце...

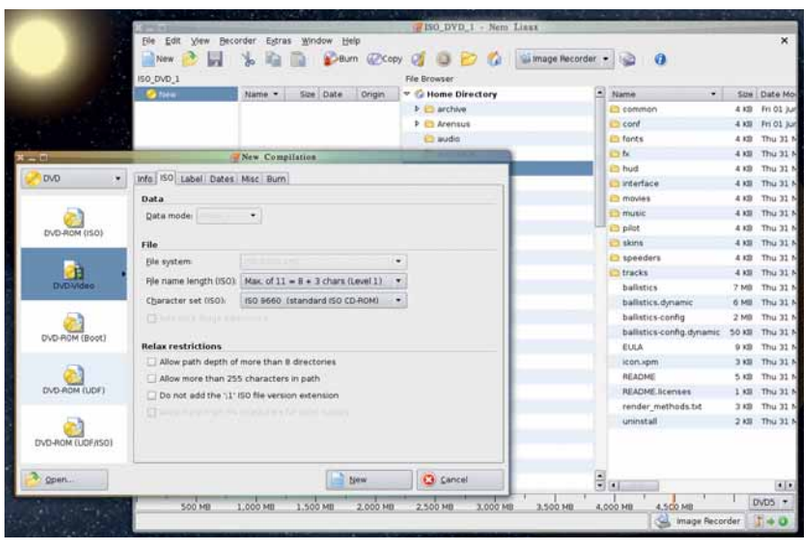
» Запись CD, DVD, дисков HD-DVD и Blu-ray без забот настройке. Свободные альтернативы: *K3b* (KDE) и *GnomeBaker* (Gnome).

Nero – одно из самых популярных приложений для Microsoft Windows, именно его многие по умолчанию используют для записи CD и DVD. С годами Windows-версия выросла в колоссальную коллекцию программ, объединяющую средства для мультимедиа-воспроизведения, трансляции медиа на другие устройства, конверсии форматов и резервного копирования. Но сердцем всего этого собрания остается записывающая программа *Nero*, которая и была портирована в Linux.

Nero Linux 3 – это третья Linux-версия, причем с каждым новым выпуском программа все больше походила на Windows-вариант, пока, наконец, не было заявлено о функциональной идентичности обеих версий. При этом *Nero* – вполне Linux-продукт. Доступны и 32-, и 64-битная версии, есть пакеты как в виде Deb-файлов, так и в формате *RPM* для инсталляции в большинстве современных Linux-дистрибутивов. Официально поддерживаются RHEL, Fedora, SUSE, Debian и Ubuntu, но и с другими проблем не ожидается. Единственное непреложное требование – наличие *GTK+ 2.4.10* и *libstdc++6 3.4*; оба пакета найдутся в любом нестаром дистрибутиве.

Жгучие вопросы

Добрая весть: новая версия собрана на *GTK2*, отчего пользовательский интерфейс перешагнул, наконец, в XXI век. Первое, что бросается в глаза – почти полное внешнее сходство *Nero Linux* с Windows-соперником. И это правильно, ведь многие новоиспеченные линуксоиды уже имеют некоторый опыт общения с *Nero*. Кроме знакомой раскладки окон и меню, в Linux перешел и тщательно отлаженный механизм Мастера записи. Зачастую это наипростейший путь создания нового диска, ведь еще до того, как в окне проекта появится его «скелет», Мастер уже проделает некоторую подготовительную работу. Например, для записи фильма на DVD *Nero* подготовит подходящую систему каталогов и настроит параметры диска.



› **Функционально *Nero Linux* идентичен Windows-версии, но для прорыва в лидеры ему надо научиться преобразованию форматов.**

А вот чего *Nero* не умеет – так это конвертировать видеофайлы для записи на DVD, а ведь это наиболее трудоемкая часть процесса. То же, к сожалению, справедливо и в отношении аудио-файлов. *K3b*, популярный (кстати, свободный) инструмент KDE для прожига CD/DVD, конвертирует аудиофайлы прямо во время перетаскивания их в список для записи. *Nero* до этого еще далеко, хотя в программе есть окно перекодировки, в котором можно оперировать форматами FLAC, Ogg, MP3 и WAV.

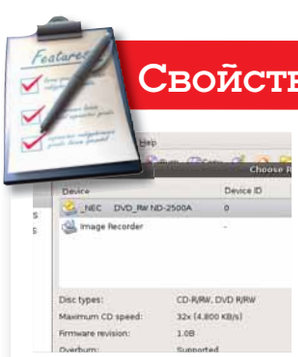
Blu-ray и HD-DVD

Несмотря на этот недостаток, для повседневной записи данных на CD или DVD *Nero* – превосходная программа. Наглядно отображается оставшееся свободное пространство, можно легко перекраивать диск и выбирать из множества доступных форматов. Уровень контроля над процессом весьма высок: есть поддержка технологии overburn (можно втиснуть на диск больше данных) и режима Disc-at-Once, для совместимости с аудиодисками. Новая версия поддерживает запись HD-DVD и Blu-ray, причем последний позволяет сохранить на единственном диске до 50 ГБ данных.

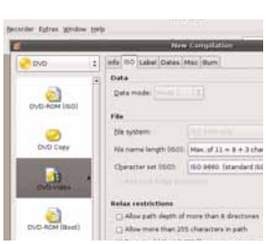
Пока эти приводы очень дороги, но ожидается, что в течение ближайших 12–18 месяцев они значительно подешевеют. Фактически, Windows-версия *Nero* приобретается при покупке нового привода. Приятно видеть Linux-альтернативу, так как она уравнивает в правах пользователей Linux с Windows-коллегами. Этот довод может стать решающим при выборе Linux-версии: свободные аналоги хороши, но – почему бы не прихватить инструментик к новому устройству?

Nero – отличный продукт: он надежен, хорошо работает и совместим с новейшим оборудованием. Да, программа платная, но если есть проблемы с *K3b* или *Gnomebaker*, то этот вариант стоит рассмотреть. LXF

Свойства навскидку



Поддержка оборудования
Без всякой мороки с настройкой, *Nero* просто находит и правильно использует имеющееся оборудование.



Мастер записи
Большинство задач можно решить парой щелчков мышью в окне Мастера записи, который «знаком» почти со всеми типами дисков.

LINUX FORMAT Вердикт

Nero Linux 3

Разработчик: Nero AH
Сайт: www.nero.com
Цена: €19.99

Функциональность	6/10
Производительность	8/10
Простота использования	8/10
Документация	7/10

» *Среди свободных программ есть и получше, но Nero, несомненно, ценное приобретение для Linux-платформ.*

Рейтинг 7/10

TwonkyMedia 4.4

Мультимедийных устройств воспроизведения – пруд пруди, а вот совместимых с ними серверов не так уж много. **Грэм Моррисон** нашел один из тех, что работают под Linux.

Вкратце...

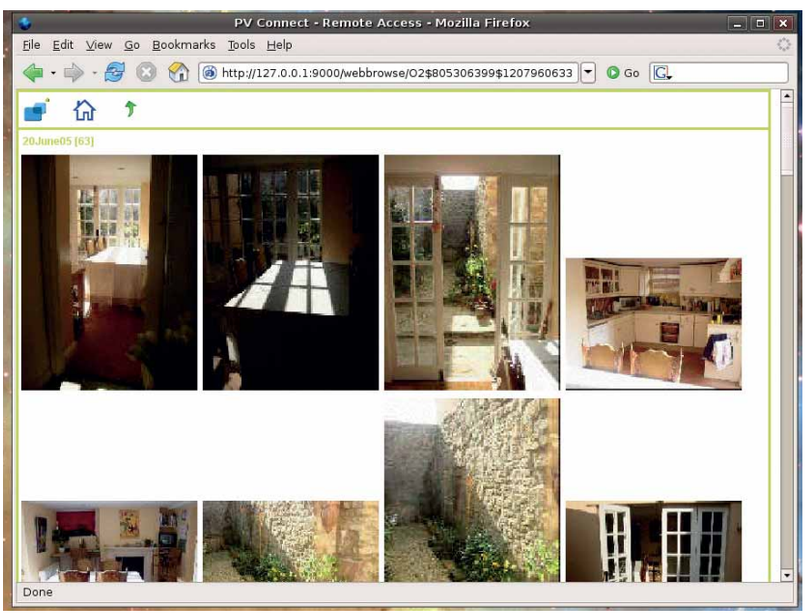
» Трансляция музыки, видео и полной домашней коллекции фото на любое UPnP-совместимое устройство. Свободные альтернативы: *Slimserver* и *MythTV*.

Universal Plug and Play (UPnP) совершил революцию в мире домашней медиа-аппаратуры. Несколько лет назад, если человек покупал устройство для трансляции музыки на hi-fi, оно могло оказаться несовместимым с аппаратом для передачи видео на телевизор из-за использования разных протоколов. UPnP изменил положение, предложив пакет стандартных сетевых протоколов, которых домашние медиа-устройства могут придерживаться в целях совместимости. А где мы обычно храним свои медиа-файлы? Один из вариантов – на Linux-компьютере: тут нужна всего лишь программа-сервер UPnP. Вот эту функцию и выполняет *TwonkyMedia*. Что замечательно – *TwonkyMedia*, одна из немногих, сертифицирована Digital Living Network Alliance, группой, которая следит за соблюдением стандарта UPnP.

Один инсталлятор для всех

Инсталляция происходит через сценарий оболочки. Если не отредактировать этот скрипт вручную, то установка по умолчанию происходит в **/usr/local**, на любом дистрибутиве. Само по себе это не столь важно, ведь *TwonkyMedia* – вполне самодостаточный пакет, но нам доводилось видеть и более утонченные инсталляторы. Процесс установки помещает стартовый скрипт в **/etc/init.d** и автоматически запускает сервер. Но, поскольку особенности дистрибутивов в расчет не принимаются, то конкретное место размещения для автоматического запуска во время загрузки скрипту неизвестно, пользователю придется вмешиваться самому.

Взаимодействие с сервером возможно только через web-интерфейс, доступный через изрядно загруженный порт 9000 (например, им пользуется *Squeezebox*, который мы рассматривали пару месяцев назад). К нашему разочарованию, способа перейти на другой порт в *TwonkyMedia* нам найти не удалось. Конечно, можно запросто сменить настройку порта в *Squeezebox*, но так ведь не полагается!



› На случай, если нет подходящего устройства воспроизведения, в web-интерфейс встроен собственный UPnP-клиент.

При входе в web-интерфейс первым делом надо ввести лицензионный ключ или запустить 30-дневную демо-версию; затем указать серверу местонахождение медиа-файлов. Можно добавить сколько угодно директорий; можно также ограничить каталог конкретным типом медиа. Это бывает удобно, если файлы в директории хранятся вперемешку. Музыкальные коллекции зачастую сопровождаются обложками музыкальных альбомов, поэтому, ограничив работу лишь файлами музыкальных форматов, можно существенно облегчить себе жизнь.

По умолчанию, медиа-коллекция каждые 60 минут сканируется в поисках новых файлов, но после первоначальной индексации ресурсы процессора на это почти не тратятся. И хотя *TwonkyMedia* находит лишь файлы известных ей форматов, их список более чем внушителен. Впечатляют возможности представления коллекций на UPnP-устройствах. Например, музыкальные произведения можно рассортировать по альбомам и исполнителям, по жанрам, рейтингу, дате и даже по иерархии папок, в которых хранятся ваши мелодии.

Без конверсии

Используя сервер *TwonkyMedia*, можно просматривать медиа-коллекции с любого домашнего UPnP-устройства. Официально поддерживаются многие модели; среди них Xbox 360 от Microsoft и PlayStation 3 от Sony, Nokia N770 и N800, а также новейшие мобильные телефоны уровня N95. Мы попробовали в работе некоторые устройства, например, PS3, PSP и *Squeezebox*. Все они работали, как и ожидалось, только вот у PS3 отсутствовали пауза, перемотка назад и быстрая вперед.

Мы сошлись на том, что самая крупная проблема – это необходимость соблюдать соответствие форматов каждому устройству, так как *TwonkyMedia* не умеет перекодировать файлы из одного формата в другой. В отличие от пользователей других ОС, выбор UPnP-серверов для Linux не очень-то велик. Сервер *TwonkyMedia* неплохо управляется с домашним медиа-хозяйством, но ему совершенно необходимо научиться конверсии файлов. И тогда мы будем рекомендовать его безоговорочно. LXF

Поддерживаемые форматы

Музыка
MP3, WMA, WAV, 3GP, M4A, MP4, LPCM, Ogg Vorbis, FLAC, MP2, AC3, MPA, MP1, AIF

Изображения
JPEG, PNG, TIF, BMP

Видео
MPEG1, MPEG2, MPEG2-TS, AVI, MPEG4, WMV, VOB, DivX, 3GP, VDR, ASF, MPE, DVR-MS, Xvid, M1V, M4V

Интернет-радио
vTuner, Shoutcast

LINUX FORMAT Вердикт

TwonkyMedia 4.4

Разработчик: TwonkyMedia
Сайт: www.twonkyvision.com
Цена: €29.95

Функциональность	5/10
Производительность	8/10
Простота использования	7/10
Цена	7/10

» *Если есть подходящая аппаратура, то это неплохой выбор. В противном случае, лучше подождать версии 5.*

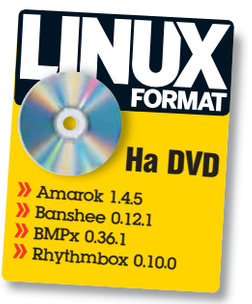
Рейтинг 7/10

Сравнение

» Каждый месяц мы сравниваем тонны ПО – а вы можете отдыхать!

Аудиоплейеры

Крис Хауэллз смахнул пыль со своих наушников и проводов с золочеными наконечниками и проэкзаменовал лучшие аудиоплейеры.



Paul Blachford

Наш выбор

Amarok с. 19
Banshee с. 22
BMPx с. 22
Juk с. 20
Rhythmbox с. 21
XMMS с. 20

Если вас бесит подлая манера компакт-дисков притягивать к себе неряшливые отпечатки пальцев и царапины, невзирая на заботливое отношение, вы обрадуетесь достойной альтернативе. Альтернатива известна уже давно: это MP3.

Способ сжатия MP3 с потерей качества в условиях весьма ограниченного свободного пространства на жестких дисках и MP3-плейерах – это объективная необходимость. Но и на MP3 свет клином не сошелся. Приверженцы Ogg Vorbis заявляют, что этот формат обеспечивает лучшее качество звучания при более высокой степени сжатия, чем MP3, что очень важно, если нужно втиснуть на устройство как можно больше музыки. Еще один вариант – FLAC (Free Lossless Audio Codec), популярный среди меломанов, отвергающих «ущербные» форматы типа MP3 и Ogg Vorbis, но желающих сохранить приемлемые размеры файлов.

Чего мы ждем от плейера?

Чтобы получить максимум удовольствия от объемистой коллекции цифрового аудио, необходимо подобрать подходящую программу для управления ею.

Многие из рассмотренных нами проигрывателей обладают функцией интеграции с портативными аудиоплейерами типа iPod, с которыми можно работать как со съемными USB-носителями. Это очень полезное качество, экономящее много времени, если музыкальное собрание на портативном устройстве часто подвергается пересмотру и обновлению.

Важное достоинство – способность к обработке тэгов метаданных (данных о данных) – например, сведений об исполнителе и дорожке, помещенных внутрь файла. Многие цифровые аудиоплейеры превосходно разбираются в тэговой информации, что существенно облегчает организацию музыкальной коллекции.

Интернет поощряет общение и взаимодействие людей друг с другом. Многие плейеры интегрируются для этих целей с сетевым сообществом на www.last.fm. Став проводником «социальной музыкальной революции», этот сайт позволяет заявить о себе новым исполнителям и объединяет людей с похожими музыкальными вкусами.

Итак, выбираем лучший аудио-плейер.

Про наш тест...

Тест проводился в Kubuntu 7.04 Feisty Fawn, установленной на Lenovo ThinkPad X60s с 1,5 ГБ ОЗУ (установлен лимит в 256 МБ) и 1,66-ГГц процессором Intel Core Duo. Программы сравнивались по следующим критериям:

» **Производительность** Мы импортировали 1 500 музыкальных файлов в форматах MP3 и Ogg Vorbis общим объемом 12 ГБ, замеряя скорость обработки и степень использования процессора.

» **Удобство использования и документация** Мы оценивали, насколько интуитивно понятен интерфейс программы и насколько качественна документация.

» **Аудиофункционал** Мы сравнивали количество и качество поддерживаемых форматов аудиофайлов, а также интеграцию с портативными плейерами, интернет-радио и т.п.

» **Обработка тэгов** Мы рассматривали способность разных плейеров обрабатывать разного рода тэговую информацию из аудиофайлов, а также возможность групповой работы с подобной информацией.

Amarok

Кто бы подумал, что мифический волк способен организовать музыкальную коллекцию?

Amarok – это лощеное KDE-приложение с девизом «открой свою музыку заново» и названием, взятым из эскимосской мифологии. Хотя фактически программа не является частью KDE, а, с учетом нужд разработчиков, выпускается по независимому расписанию. К тому же для вывода аудио *Amarok* использует *Xine*, а не ARTS (Analogue Real Time Synthesizer), которым обычно пользуются KDE-приложения. Это позволяет плейеру реализовывать некоторые приятные функции, например, плавный переход между дорожками, когда в конце одной темы звук плавно затихает, а в начале следующей – так же плавно нарастает.

Amarok – дружелюбная программа с развитой справочной системой. При первом запуске *Amarok* предлагает создать музыкальную коллекцию из всех найденных музыкальных файлов. В процессе индексации программа читает каждый файл и извлекает из него мета-данные, которые затем использует для построения плей-листов.

Списки воспроизведения

Все функции *Amarok* доступны через вкладки, корешки которых располагаются с левой стороны главного окна. Вкладка Context (Контекст) служит для вызова контекстного браузера, Playlists (Плейлисты), понятно, выводит доступные списки воспроизведения. Вкладка Files (Обзор) позволяет просматривать файловую систему в поисках дополнительных музыкальных файлов, а открыв вкладку Magnatune, можно купить избранное музыкальное произведение в магазине Magnatune. Наконец, вкладка Devices (Устройства) позволяет взаимодействовать с портативными аудиоплейерами.

Одна из особенностей *Amarok* – это контекстный браузер. Он предоставляет просто фантастический объем информации, связанной с конкретным произведением. Например, можно просмотреть обложку альбома или сведения о том, насколько часто воспроизводилась эта дорожка прежде, а также быстро перейти к другим дорожкам или альбомам данного исполнителя. Все эти функции превосходно организованы и приятно оформлены визуально.

› *Amarok* поддерживает протокол DAAP для трансляции музыки с iTunes и аналогичных серверов.

› С помощью Диспетчера обложек внешний вид *Amarok* нетрудно изменить по своему вкусу.

Хорошо продумана организация плейлистов. В базу данных плейера можно импортировать собственные статические списки в формате M3U, но более интересны динамические списки, составляемые различными способами. Хотите прослушать свои любимые дорожки или узнать, какую песню любите меньше всего? Для *Amarok* это не проблема. Трансляция интернет-радио тоже доступна через вкладку Плейлисты. Некоторые станции предустановлены, а добавить новые совсем нетрудно.

При желании послушать песни какого-либо исполнителя можно отправляться прямо во вкладку Collection (Коллекции), где находится упорядоченный список всех исполнителей/альбомов, которые есть в музыкальной коллекции; там же можно добавить все, что захочется, прямо в плейлист. Через контекстный браузер можно просмотреть слова песни и сведения об исполнителе, взятые из Wikipedia.

Импорт музыки

Неплоха поддержка музыкального магазина Magnatune. Функция встроена в *Amarok*, поэтому можно прослушать демо-ролики и, если понравится, тут же приобрести их.

Если музыкальное содержимое портативного медиа-плейера часто меняется, интеграция с *Amarok* сэкономит вам кучу времени. Программа обладает модулями поддержки самых разных устройств, включая iPod, плейеры от Creative и iRiver и стандартные USB-носители. Можно создать очередь песен для отправки на портативное устройство.

Amarok прекрасно поддерживает графическое оформление альбомов, управлять которым можно через окно под названием Cover Manager (Диспетчер обложек): либо загружать обложки с Amazon, либо присвоить каждому альбому собственную обложку из локального хранилища изображений. При воспроизведении дорожки *Amarok* может вывести обложку альбома как в контекстном браузере, так и контекстном меню системного лотка.

Программа обладает даже возможностью прожига CD через *K3b*, записывающее приложение KDE. Можно создавать стандартные музыкальные компакт-диски, а можно записывать файлы MP3 на CD с данными.

Музыкальная трансляция

Amarok угодил и любителям наблюдать при прослушивании музыки яркие видеоэффекты. Здесь есть разнообразные режимы визуализации, используемые для генерации графики в такт звучания песни.

И для любителей автоматизации у программы есть несколько козырей в рукаве. Один из них – полная совместимость с DCOP (Desktop COmmunications Protocol), с помощью которого *Amarok* можно управлять из других программ и скриптов.

Функциональность проигрывателя легко расширить с помощью диспетчера скриптов. Некоторые готовые скрипты распространяются вместе с программой, многие загружаются через Интернет. Сделать это совсем нетрудно, ведь в программу встроен браузер скриптов, который берет на себя их поиск в Сети и загрузку.

Обладая богатым набором функций и тщательно продуманным интерфейсом, *Amarok* устанавливает весьма высокие стандарты, равняться на которые другим плейерам совсем не просто.

LINUX FORMAT Вердикт

Amarok

Версия: 1.4.5
Сайт: http://amarok.kde.org
Цена: Бесплатно под GPL

» *Amarok – замечательная программа, великолепно задуманная и безупречно выполненная.*

Рейтинг 10/10

XMMS

‘X’, видимо, означает «кроссплатформенный».

Если попросить нескольких линуксоидов перечислить аудиоплейеры для Linux, то *XMMS*, X MultiMedia System, непременно будет присутствовать во всех ответах. *XMMS* с нами уже много лет, но именно его возраст указывает на явный застой в его развитии.

Интерфейс пользователя *XMMS* состоит из маленького окошка, напоминающего устройство hi-fi. К сожалению, окошко очень маленькое даже при скромном разрешении 1024x768. Нажатием клавиши D оно удваивается в размерах, но в этом случае стандартная «шкурка» становится совсем уж неприглядной.

Внешность плейера можно изменить с помощью различных тем – *XMMS* поддерживает и темы медиаплейера Winamp 2 для Windows, и «шкурки» собственного формата. Некоторые пользователи разработали собственные темы, загрузить которые можно с сайта *XMMS*. Там же можно найти некоторые дополнительные модули.

Интерфейс весьма скуден и неприветлив. Например, чтобы открыть окно списка воспроизведения, нужно нажать кнопку с надписью ‘PL’, а кнопки для проигрывания в случайном порядке и повторения композиции названы просто ‘RAND’ и ‘REP’.

Поддержка плей-листов в *XMMS* рудиментарна. Загрузка готовых списков M3U и простейшие манипуляции над ними вроде воспроизведения в случайном порядке – это почти все, что умеет программа. При желании файлы можно добавлять напрямую в браузер, и тогда необходимость во всяких списках вообще отпадает.

Основные функции *XMMS* могут быть расширены с помощью дополнительных модулей – например, поддержка каждого формата музыкальных файлов выполнена в виде модуля. *XMMS* обладает разнообразными модулями визуализации для сопровождения музыки приятными видеоэффектами.

Есть простейший редактор метаданных, но он годится лишь для косметических изменений в небольшом количестве файлов, поэтому, если коллекция велика, а изменить нужно много, то лучше поискать нечто более пригодное.

Первая версия *XMMS*, названная X11Amp, увидела свет в далеком 1997, но в последние годы разработка встала на якорь.

LINUX FORMAT Вердикт

XMMS

Версия: 1.2.10
Сайт: www.xmms.org
Цена: Бесплатно под GPL

XMMS выглядит удручающе и не выдерживает никакого сравнения с соперниками.

Рейтинг 2/10

Juk

Как вы поняли, это KDE-приложение.

Скотт Уилер [Scott Wheeler] приступил к разработке *Juk* в 2000 г.; первоначально проект назывался *QTagger*. В 2002 программа вошла в CVS-дерево KDE, где и продолжила свой рост. Пользовательский интерфейс *Juk* напоминает *Jukebox*, а значит, предназначен для эффективной обработки многочисленных плей-листов.

При первом запуске программа просит указать ей музыкальную коллекцию. В процессе индексации *Juk* просматривает всю коллекцию, наполняя список воспроизведения сведениями об исполнителях и названиях композиций. Если во время сканирования программа встретит плей-листы в формате M3U, они тоже будут добавлены к общей коллекции и появятся в левой области основного окна.

Приятно экспериментировать с созданием списков воспроизведения. Интересна функция Playlist From Folder (Плейлист из папки) – нужно просто указать папку для создания списка, и все ее содержимое будет добавлено в новый список.

Значок *Juk*, встроенный в системный лоток, позволяет контролировать различные аспекты работы программы: воспроизведение, паузы и переход к другой дорожке. Ненавязчивую всплывающую подсказку можно настроить на отображение в системном лотке сведений о текущей дорожке и показ обложки альбома. Еще одна приятная мелочь: наводя курсор мыши и прокручивая ее колесико, можно переходить к следующей или предыдущей дорожке.

У *Juk* есть и свой диспетчер для упорядочения небольших изображений альбомных обложек, хотя он не так удобен и эффективен, как в *Amarok*.

Воспроизведение аудио – лишь часть обширного набора функций *Juk*: например, здесь есть хорошо организованные инструменты для работы с тэгами. Если в коллекции есть файлы с правильными названиями, но недостаточной метаинформацией, то можно воспользоваться функцией извлечения метаданных из названий файлов. С другой стороны, имея полноценные метаданные, но неполные имена файлов, можно попытаться использовать функцию переименования файлов.

«Если у вас есть списки M3U, то они тоже попадут в коллекцию».

Изображения могут быть закреплены за конкретными дорожками и отображаться при их воспроизведении.

LINUX FORMAT Вердикт

Juk

Версия: 2.3.3
Сайт: http://developer.kde.org/~wheeler/juk.html
Цена: Бесплатно под GPL

Juk стоит внимания благодаря умелой работе с тэгами. А в придачу – неплохой аудиоплейер.

Рейтинг 7/10

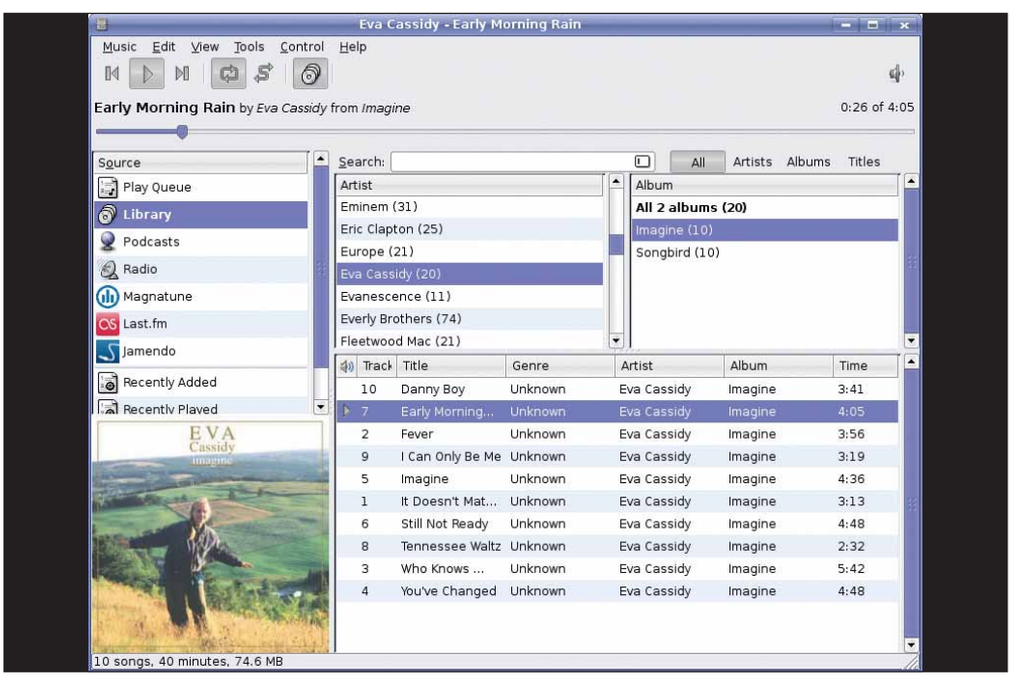
Rhythmbox

Gnome тоже любит музыку…

Rhythmbox – это Gnome-приложение в духе *iTunes*. Установить и запустить его нетрудно: в первый раз программа запускается в режиме Мастера. Мастер сформирует и настроит музыкальную коллекцию, с которой в дальнейшем и будет работать *Rhythmbox*. Программа просмотрит предложенную коллекцию и сформирует библиотеку для работы с ней.

Интерфейс пользователя *Rhythmbox* состоит из четырех основных областей: списка источников, списка исполнителей, списка альбомов и списка дорожек. Список дорожек может быть ограничен рамками одного или более исполнителей и/или альбомов. В верхней части окна имеется строка поиска.

Обложки альбомов автоматически загружаются из Amazon и отображаются в левой части главного окна при воспроизведении соответствующей композиции. К нашему разочарованию, без запуска файлового менеджера оказалось невозможно вручную закрепить за конкретными треками те обложки, которые *Rhythmbox* не загрузил из Интернета.



› Интерфейсы *Rhythmbox* и *Juk* схожи. Органы управления расположены вверху слева, а индикатор положения на дорожке – в верхней части экрана.

Риппинг и прожиг

Мы были поражены, увидев, как *Rhythmbox* автоматически, самостоятельно «засек» ресурсы *iTunes* в нашей локальной сети. Для трансляции аудио по сети *iTunes* использует протокол DAAP (Digital Audio Access Protocol), поэтому, имея хотя бы один компьютер с запущенным *iTunes* или любой другой DAAP-сервер, можно использовать *Rhythmbox* для приема музыки.

Модульная архитектура позволяет *Rhythmbox* легко расширять свой набор функций. Существуют модули для поддержки last.fm, Magnatunes и загрузки текстов песен.

Rhythmbox поддерживает CD-риппинг и запись дорожек на аудио-диски, но реализованы эти функции не идеально. Риппинг дисков осуществляется с помощью Gnome-приложения *Sound Juicer*, но мы, как ни старались, не нашли в меню ни одной зацепки для осуществления этой функции. В конце концов мы поняли, что *Rhythmbox* просто не отображает пункты для управления этим процессом, пока не увидит CD-привод, оставляя нас лишь гадать, в чем дело. Нам кажется, что куда удобнее было бы обозначить присутствие функции независимо от режима. А когда риппинг невозможен, значок или пункт меню можно просто дезактивировать, переведя в нечувствительное состояние.

› Первые шаги после запуска *Rhythmbox* направляет мастер.

Разочаровало нас и обращение *Rhythmbox* с сообщениями об ошибках. Например, при попытке воспользоваться модулем Magnatune нас ждало сообщение о невозможности установки модуля, без всяких объяснений, и мы даже не могли обратиться за помощью в Сеть или на Google, чтобы посмотреть, не сталкивался ли кто-нибудь еще с подобной проблемой. Только при просмотре сообщений в консольном выводе мы увидели, что причина крылась в ошибке Python, сообразили, что не хватает компонентов Glade для Python, и исправили ошибку. Но нам кажется, что получить столь простую функцию такой ценой – слишком накладно.

Rhythmbox поддерживает просмотр новостей, для чего нужно ввести URL нужной ленты RSS в соответствующем диалоговом окне, совершенно несложном в обращении. Если у вас есть iPod, *Rhythmbox* может взаимодействовать с ним благодаря специальному модулю; владельцам других устройств повезло меньше.

Визуализация организована неплохо, при воспроизведении анимация появляется на месте списка дорожек. В идеале, возможность установки разных вариантов визуализации могла бы быть проще – по умолчанию доступен лишь один, а инструкции по установке других вариантов отнюдь не очевидны.

«В режиме Party Rhythmbox станет полноэкранным».

В режиме Party *Rhythmbox* становится полноэкранным, а в режиме Small Display представлены только основные органы управления.

LINUX FORMAT Вердикт

Rhythmbox

Версия: 0.10.0
Сайт: www.gnome.org/projects/rhythmbox/
Цена: Бесплатно под GPL

» *По части функциональности это нечто среднее между Juk и Amarok. Неплохой выбор для пользователей Gnome.*

Рейтинг 8/10

BMPx

Случай «непростого» прослушивания.

BMPx ведет свою родословную от *XMMS*, хотя, глядя на его интерфейс, этого не скажешь: за прошедшие годы переделок от оригинального кода *XMMS* мало что осталось.

BMPx не так дружелюбен, как большинство тестируемых плейеров: он единственный заставил нас скрести в затылке в поисках возможности «затолкать» в него нашу музыкальную коллекцию. Прочесав все многочисленные вкладки, мы установили, что коллекцию следует импортировать во вкладке Album через меню Albums.

Затруднения вызвала и реализация некоторых функций *BMPx*, например, обращение с обложками альбомов. В программе есть пункт меню под названием Fetch Covers For All Albums [Достать обложки для всех альбомов], но его выбор ни к чему не привел. Досадно, что мы не увидели ни сообщений об ошибках, ни объяснений причин отказа этой функции.

Теперь о приятном. *BMPx* поддерживает риппинг, запись и конверсию аудио-дисков в несколько форматов, включая MP3. Поддержка last.fm в программе также присутствует, наряду с прослушиванием потоковых серверов Icecast и Shoutcast. Эти функции, открывающие доступ к значительной части интернет-вещания, доступны через значки в левой части окна.

Поддержка плей-листов в *BMPx* хуже, чем в других протестированных программах. Дорожку можно добавить в список, но после этого список придется вручную сохранить, а затем добираться до него по файловой системе, чтобы открыть снова. Стабильность никуда не годится: *BMPx* завоевал сомнительную честь быть единственным плейером, рухнувшим во время теста.

Интерфейс намеченной к выпуску версии *BMPx 0.40* чище и удобнее в использовании и выглядит заметно усовершенствованнным, и мы надеемся, что обновленная версия плейера будет стабильной.

› При всех стараниях нам так и не удалось добиться поддержки альбомных обложек в *BMPx*.

> «BMPx — единственный плейер, который рухнул во время теста».

LINUX FORMAT Вердикт

BMPx

Версия: 0.36.1
Сайт: http://bmpx.beep-media-player.org/
Цена: Бесплатно под GPL

» При нехватке функциональности и удобства, BMPx потребляет сравнительно мало ресурсов.

Рейтинг 5/10

Banshee

Один из многих: боец, но не лидер.

Расписанный как «система управления и воспроизведения музыкальных файлов для Gnome», *Banshee* показывает класс уже при запуске. Следуя его предложениям, мы указали местоположение файлов, и *Banshee* занялся составлением коллекции.

Интерфейс пользователя хорошо продуман; во время воспроизведения дорожек альбомная обложка отображается в левом нижнем углу окна. Рядом с обложкой расположена панель с рекомендациями по выбору модных исполнителей и композиций на основе информации из базы данных last.fm.

Banshee любит хвалиться способностями к управлению музыкальными коллекциями, но мы нашли их не более чем адекватными. Работа с тэгами организована неплохо, можно обрабатывать тэги нескольких файлов одновременно, а вот переименовывать файлы или перемещать их на диске невозможно. Крайне бесит невозможность импортировать плейлисты в формате M3U.

Эта программа – не только аудиоплейер, в ней возможны риппинг и запись CD, а также преобразование файлов в различные форматы, включая AAC, MP3, Ogg и FLAC. Все это реализовано довольно просто, без путаницы с излишними функциями.

Модули – важная часть *Banshee*: большей частью своей функциональности он обязан именно модульной архитектуре. На сайте *Banshee* имеются модули от сторонних разработчиков, но в будущем мы надеемся увидеть встроенный в программу браузер для их поиска и загрузки.

Создавать новые плейлисты несложно, их названия сразу же появляются в левой части окна программы. Интересна функция «интеллектуального» создания списков воспроизведения, когда они генерируются автоматически, заданием одного или более критериев выбора дорожек.

Любители послушать интернет-радио наверняка заметят его надежную поддержку в *Banshee*. Несколько радиостанций включены по умолчанию, добавить другие не составляет большого труда.

› К сожалению, в этой версии *Banshee* невозможен импорт списков воспроизведения M3U.

> «Любители интернет-радио заметят его надежную поддержку в Banshee».

LINUX FORMAT Вердикт

Banshee

Версия: 0.12.1
Сайт: http://banshee-project.org/
Цена: Свободно под GPL

» У Banshee немало достоинств, но в наши дни неспособность к импорту списков воспроизведения непростительна.

Рейтинг 6/10

Аудиоплейеры

Вердикт

Amarok 10/10

Некогда – не так давно – пользователи довольствовались загрузкой в аудиоплейер списка дорожек да их воспроизведением, останавливая их, переходя с дорожки на дорожку… вот, собственно, и все. Сегодня от плейера требуется гораздо больше: риппинг и прожиг CD, интеграция с портативными устройствами и многое другое.

Сравнение десятилетнего *XMMS* с современными прогрессивными плейерами вроде *Amarok* или *Rhythmbox* было, вероятно, не совсем корректным, но в нем наглядно отразился прогресс, достигнутый в данной области за последние несколько лет.

Amarok по праву занял первое место в этом сравнении: нас поразили его функциональность, лоск, дружелюбие и тщательно продуманный дизайн. Мы не смогли найти изъяна ни в одной из функций *Amarok*, потому и присвоили ему десять из десяти возможных очков.

Amarok разрабатывается активной группой способных программистов, которая регулярно публикует сведения о ходе процесса разработки по адресу http://dot.kde.org.

Работа над версией 2.0 уже идет. Обещано усовершенствовать интерфейс и придать контекстному браузеру еще более заметное положение в программе. Обещано также, что *Amarok* сможет работать на других платформах, Windows и OS X – добрая весть для наших иноплатформенных друзей.

Rhythmbox – второй плейер в нашем рейтинге. Ему не достает совершенства, свойственного *Amarok*, но он все же уверенно справляется с воспроизведением музыки, а также с некоторыми сопутствующими задачами (например, риппингом и прожигом CD). Производительность программы, однако, оставляет желать лучшего: импорт музыкальной коллекции занял долгие 13 минут.

› Создание аудио-CD с помощью *K3b* – лишь одна из множества функций *Amarok*, и не зря программа победила в нашем тестировании.

Третье место занял *Juk*. Способности *Juk* в области обработки метаданных оказались одними из лучших в нашем Сравнении, хотя у программы есть кое-какие недостатки в других областях, таких как синхронизация музыкальных коллекций с портативными плейерами, трансляция интернет-радио и поддержка last.fm.

Стоит упомянуть, хотя и не в хвалебном тоне, и о *XMMS*. Эта программа, похоже, приближается к своему закату, и это единственный плейер, которым мы всячески избегаем пользоваться ввиду скудости интерфейса и слабой функциональности. Застой в разработке на протяжении последних лет привел к тому, что программу затерли более свежие, современные и активные соперники. LXF

Ваше мнение

Предпочитаете ли вы слушать музыку на hi-fi, или на компьютере? Каким ПО вы пользуетесь для управления своей крупной медиа-коллекцией? Сообщите о своем опыте и пожеланиях на letters@linuxformat.ru.

Таблица функций

Приложение	Импорт музыки (сек)	Использование процессора (%)	MP3	Ogg	FLAC	Загрузка плей-листов	Трансляция интернет-радио	CD-риппинг	CD-прожиг	MP3-плейер
Amarok	113	1	✓	✓	✓	✓	✓	✓	✓	✓
Banshee	73	10	✓	✓	✓	☒	✓	✓	✓	✓
BMPx	108	10	✓	✓	✓	✓	Только Shoutcast/Icecast	✓	☒	☒
Juk	56	0.5	✓	✓	✓	✓	☒	☒	☒	☒
Rhythmbox	773	10	✓	✓	✓	✓	✓	✓	✓	Только iPod
XMMS	нет	0.5	✓	✓	Через модуль расширения	✓	✓	☒	☒	☒

Distrowatch

Ежемесячная сводка новостей дистрибутивов Linux

ЛАДИСЛАВ БОДНАР
основатель, начальник, редактор и сотрудник DistroWatch.com.

IP-рэкет

Недавно Xandros и Linspire заявили о соглашениях по защите интеллектуальной собственности [intellectual property, IP] с Microsoft. Это было ответом на последние заявления софтверного гиганта о нарушении некоторых его патентов со стороны Linux, что на каком-то этапе может привести к юридическому преследованию пользователей Linux.

Линус Торвальдс расценил эти выпады как не более чем FUD-кампанию (Fear, Uncertainty, Doubt – страх, неуверенность, сомнение), а Марк Шаттлворт из Ubuntu назвал их «рэкетированием». Так почему же некоторые Linux-деятели так стремятся влезть в одну лодку с компанией, продолжающей нападки на Linux и ее пользователей?

Одно из возможных объяснений – при отсутствии информации о денежной стороне сделок – это их отчаянная жажда рекламной шумихи. Несмотря на то, что Xandros и Linspire существуют уже более пяти лет, они так и не сумели привлечь мало-мальски приличное число пользователей.

Верный путь

В отличие от них, Ubuntu, OpenSUSE и Fedora – весьма успешные проекты. Вместо поиска возможностей попасть в заголовки газет они выбрали другой путь: заинтересовали сообщества своих пользователей и открыли проекты для сторонних участников.

Заключение сомнительных пактов об интеллектуальной собственности – плохой способ продвижения наверх. Лучше не заключать сенсационные сделки, а сфокусироваться на свойствах, присущих свободному ПО: совместной работе, открытости, инновациях и акценте на сообществе пользователей. Пренебрегая преимуществами этих уникальных свойств, все равно останешься Linux-компанией второго сорта, сколько громких соглашений ни подписывай.

ladislav.bodnar@futurenet.co.uk

Шестеро лучших

Операционные системы Хронология инноваций и идей, которые помогли сформировать мир дистрибутивов Linux.

С времен выпуска Slackware Linux 1.0 в июле 1993 года, мир дистрибутивов Linux стремительно развивался. Мы видели не только постоянно улучшающиеся релизы с обновляемыми пакетами, но и являлись свидетелями подлинно новых идей, меняющих наши представления об этой альтернативной ОС. Давайте кратко рассмотрим наиболее интересные идеи в истории дистрибутивов Linux.

Вероятно, первым большим нововведением, продвинувшим Linux в массы, была система управления пакетами RPM, представленная Red Hat в 1995 г. Неожиданно кончились дни инструментов, не учитывающих зависимости между программами и их библиотеками, и появился новый мощный способ управления ПО в Linux. По иронии, эта утилита позднее породила термин «ад зависимостей», но в момент создания она выглядела как радикальное улучшение *pkgtools* от Slackware.

В 1996, Caldera Systems выпустила Caldera Network Desktop 1.0 – этот дистрибутив Linux первым представил систему установки в стиле «выбери и щелкни». До него, большинство установок Linux выполнялись или путем ввода загадочных команд в терминале, или при помощи текстовых меню. Графический установщик Caldera был настолько хорош, что одно из его ответвлений, Lycoris Desktop/LX, использовало его до 2004 года.

> «Linux стала самой инновационной ОС в мире.»

Первой крупной компанией, вступившей на арену Linux, стала Corel в 1999 г. Поставив на то, что Linux скоро проломит брешь в господстве Windows, Corel выпустила продукт, в то время считавшийся наиболее дружественной к пользователю настольной ОС Linux. К сожалению, он не принес значительных доходов, и два года спустя фирма свернула свою Linux-деятельность.

Дистрибутив, которому Linux больше всех других обязан признанием – это знаменитый KNOPPIX LiveCD, полноценная ОС, запускаемая прямо с CD, которая была представлена в 2003 г. Хотя это был не первый подобный продукт, KNOPPIX стал популярным благодаря отличному автоматическому распознаванию устройств. Идея стартовала, и теперь мы имеем на рынке сотни LiveCD на любой вкус и для любых целей.

› Caldera OpenLinux щеголяла прекрасным графическим установщиком, но он исчез, когда пропал United Linux.

Благодаря огромной популярности технологии LiveCD, не замедлил появиться и комбинированный LiveCD с возможностью установки. Проект Mepis был первым, представившим концепцию и в 2004 г. выпустившим установочный Mepis LiveCD, который позволял пользователю перед установкой протестировать дистрибутив в Live-режиме. Сейчас установочные LiveCD предоставляют все основные дистрибутивы, за исключением OpenSUSE и Slackware.

В 2006 г. настольный Linux был буквально революционизирован аппаратным ускорением *Xgl* от Novell и *AIGLX* от Red Hat. Малоизвестный проект под названием Kororaa первым предоставил полноценный 3D-рабочий стол на LiveCD. Хотя мнения об удобстве использования этих эффектов весьма разнятся, 3D-рабочий стол Kororaa несколько месяцев являлся одним из наиболее загружаемых Linux.

Это лишь часть инноваций, которые мы видели за последнее десятилетие или около того; а будут ли еще? Несомненно! Раз большинство идей рождается в умах талантливых одиночек, то без опаски можно сказать, что Linux стала самой инновационной настольной операционной системой в мире!

Рывок Slackware

Slackware 12.0 Наконец-то дистрибутив перешел на ядро 2.6… последним!

Slackware Linux всегда был известен скорее своей стабильностью и надежностью, чем авантюрным включением последних технологий Linux. Версия 11, вышедшая в октябре 2006 г., подтвердила эту тенденцию – это был единственный из основных дистрибутивов, поставлявшийся с ядром 2.4, web-сервером *Apache 1.3* и загрузчиком *LILO* по умолчанию. Большинство обзоров расценило его как чересчур консервативный, что негативно отразилось на дистрибутиве.

Однако когда Патрик Фолькердинг [Patrick Volkerding] обновил текущий список изменений 17 марта 2007 г., стало ясно, что основатель Slackware прислушался к критике и что готовящийся релиз будет куда более современным продуктом.

Ядро Linux по умолчанию обновилось до последней версии 2.6, и то же произошло со всей цепочкой – ядро и все пакеты теперь собираются в более безопасном GCC4, а *glibc* обновлена до версии 2.5. Старая монолитная оконная система X заменена модульной *X.Org 7.2*.

Slackware Linux 12.0 с легковесным, но мощным рабочим столом Xfce 4.4.1.

Первый релиз-кандидат Slackware 12.0 был представлен в июне. Пользователи настольных версий порадуются стандартному набору современных настольных приложений; тех, кто запускает его на сервере, может насторожить размах внутренних реформ. Но уж если самый консервативный дистрибутив на планете перешел на ядро 2.6 и сопутствующие ему технологии, не является ли это знаком, что ядро созрело как стабильный и надежный продукт?

www.slackware.com

Изобилие игр

LinuX-gamers Новый LiveDVD предлагает несколько хороших 3D-игр для Linux.

Как мы все знаем, по части игр Linux считается второсортным. А так ли это?

Новый проект LinuX-gamers.net заявляет, что это миф. Он предоставляет не только свежую информацию об играх в Linux, но также и великолепный LiveDVD с коллекцией прекрасных 3D-игр.

Загрузочный LiveDVD LinuX-gamers переносит пользователя прямо на свой рабочий стол, на основе Fluxbox. Это не обычный дистрибутив Linux, поскольку он не содержит каких-либо стандартных настольных приложений (нет даже web-браузера). Однако DVD, основанный на Arch Linux, прекрасно справляется с обнаружением устройств, а также автоматической настройкой сети для доступа к игровым серверам.

На момент написания статьи, DVD предлагал следующие новые 3D игры: *BzFLag*, *Enemy Territory*, *Glest*, *Nexuiz*, *Sauerbraten*, *Torcs*, *Tremulous*, *True Combat*: *Elite*, *Warsow* и *World Of Padman*. Остальные добавления планируются в стабильном релизе 1.0, а разработчики обещают также «семейное» издание с играми для всех возрастов. Нечего и говорить, что проприетарные Linux-драйвера ATI и Nvidia также прилагаются и загружаются по умолчанию.

LinuX-gamers 0.9 был первым публичным релизом этого превосходного игрового DVD.

Хотя выбор игр явно поменьше, чем у других платформ на рынке, LinuX-gamers LiveDVD обеспечит недели серьезных развлечений – даже для продвинутых игроков. Он доступен: его можно свободно скачать с **http://live.linux-gamers.net**. LXF

3D-рабочие столы

Если вам по душе сглаженная графика, вращающиеся кубы и захватывающая анимация, то вы, вероятно, предпочитаете дистрибутивы с технологиями трехмерных рабочих столов. В таблице ниже приведен список различных дистрибутивов и их настольных 3D-компонентов, включая проприетарные драйвера Nvidia и ATI. Хотя некоторые дистрибутивы не предоставляют эти компоненты, зачастую вы можете получить их из сторонних репозиториев, таких как livna.org (Fedora), напрямую от разработчика или издателя пакета (OpenSUSE) или из нестабильного репозитория проекта (Gentoo).

Дистрибутив	Beryl	Compiz	Metisse	NVIDIA	ATI
Ubuntu 7.04	—	0.3.6	—	1.0-9755	8.34.8
openSUSE 10.2	—	0.2.2	—	—	—
Fedora 7	—	0.3.6	—	—	—
Mandriva 2007.1	0.2.0	0.3.6	0.4.0-rc4	1.0-9755	8.34.8
Debian GNU/Linux 4.0	—	0.2.2	—	1.0-8776	8.28.8
Gentoo Linux 2007.0	—	—	—	1.0-8776	8.35.5
PCLinuxOS 2007	0.2.0	0.4.0	0.4.0-rc4	1.0-9755	8.36.5
SabayonLinux 3.4-beta2	0.2.1s	0.5.2	0.4.0-rc4	1.0-9755	8.36.5
SimplyMEPIS 6.5	0.2.0	—	—	1.0-9755	8.31.5
Slackware Linux 12.0	—	0.5.0	—	—	—

Хит-парад дистрибутивов

10 самых посещаемых страниц на DistroWatch.com со 2 июня по 1 июля 2007 г.
(среднее число визитов в день)

Дистрибутив	Число визитов	
1 PCLinuxOS	2,757	‹›
2 Ubuntu	2,110	‹›
3 Fedora	1,737	‹›
4 OpenSUSE	1,275	‹›
5 Mint	1,044	↑
6 Sabayon	959	↓
7 Debian	870	↓
8 Mepis	759	↑
9 Slackware	650	–
10 Mandriva	630	–

» **DistroWatch.com** оценивает популярность дистрибутивов, основываясь на числе визитов на страницу каждого дистрибутива. Хотя это не дает представления о действительном числе установок, но показывает, какие дистрибутивы были более востребованы за данный отрезок времени.

Linux: ВЫВОДИМ ШЛАКИ

» System Rescue CD 0.3.7
» GParted 0.3.3

Linux безопасен и стабилен, только если держать его на строгой диете официальных пакетов и обновлений, чего большинство из нас не делает. **Грэм Мориссон** чистит систему от обломков без урона для важных данных.

Прежде всего подчеркнём: мы не считаем Linux нестабильным. Существуют Linux-сервера, годами работающие без единой перезагрузки. Но для пользователей ПК ситуация немного отличается. Проблема в том, что мы любим устанавливать всякие «примочки». Много «примочек». Достаточно взглянуть на содержание HotPicks либо LXF DVD, чтобы увидеть: одна из главных причин, почему люди используют Linux – огромное количество библиотек с программами для установки. И если нужное вам приложение еще не разработано, на свете много пользователей, готовых написать свое. В итоге среднестатистическую Linux-машину ждет кончина. Конечно, это случится не в первые шесть месяцев, а то и не в первый год, но неминуем момент, когда система начнет разваливаться. Засорение вашей системы обломками беспорядочно устанавливаемых в течение двух лет пакетов приведет к нарушению ее отлаженного функционирования.

Ставим диагноз

Снижение производительности обычно является первым признаком недомогания вашей системы. Если ваш рабочий стол стал дольше «проявляться» или процесс загрузки повисает при добавлении новых участников в стартовую последовательность, ваша система может «тормозить» из-за большого количества приложений или серверов, запускаемых при загрузке. Вернейший признак болезни – нестабильность системы. Если запуск приложений приводит к их краху, то налицо конфликт в программном обеспечении. Проблема в том, что очень немногие из нас придерживаются пакетов программ, предоставляемых официальным поставщиком дистрибутива. И на это есть веские основания: в официальных репозиториях редко встретишь пакеты для новых программ, их новые версии и обновления. В большинстве дистрибутивов предпочитают придерживать новые дополнения, используя их в качестве стимула для обновления системы на последний «великолепный» релиз, особенно если за него нужно платить деньги. В итоге многие из нас вынуждены прибегать к пакетам, собранным доброхотами из сообществ. К сожалению, пакеты «от третьих лиц» нередко имеют недостатки либо из-за ссылки на другие нестабильные пакеты, либо из-за отсутствия контроля качества содержащегося в них ПО.

Еще хуже дело обстоит, если вам нужно собирать программное обеспечение самостоятельно, поскольку в большинстве дистрибутивов это приведет к повреждению базы зависимостей, используемой вашим менеджером пакетов. К проблемам приводит заключительная часть процесса инсталляции, уже после компиляции, когда вы набираете **make install** от лица системного администратора.

Это дает карт-бланш скрипту установки, который теперь имеет право копировать и перезаписывать файлы в вашей системе. Новые библиотеки заменяют старые, и хрупкое равновесие зависимостей вашего менеджера пакетов нарушается. Проблема зависимостей может привести к прекращению функционирования приложений, но не должна вызывать крах вашей машины. Если же крах происходит, то обычно это указывает на проблемы с ядром, вызванные установленными вами драйверами или заплатками. Драйверы ATI и Nvidia «славятся» созданием проблем такого рода, и выявить их непросто. Тут может помочь обновление ядра, но оно способно заодно ударить по стабильности оставшейся части системы.

В подобных случаях выявлять и лечить каждую «болячку» обычно непродуктивно. Простая переустановка ОС решит большинство проблем. Но типичный пользователь Linux расценивает это как выплескивание с водой и младенца – спасшись от нестабильности, вы зато потеряете все ваши нажитые тяжелым трудом конфигурации, установленные темы и настройки прикладных программ. Ответ один: зарезервируйте нужные вам вещи, и можно отдыхать. Но что нужно сохранить? Где эти вещи живут, и как безболезненно переустановить Linux без потери ценных данных? На эти вопросы мы и попытаемся ответить на следующих страницах.

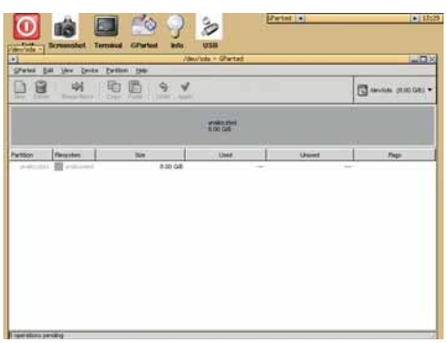
Даже если переустановка Linux кажется вам крайней мерой, мало кто из нас долго пользуется одним и тем же дистрибутивом – можно замаскировать переустановку под необходимость попробовать новый дистрибутив или изменить настройки вашей системы. Переустановка – прекрасный предлог сделать все правильно. Можно, например, рассмотреть переход на LVM (Logical Volume Manager), менеджер логических томов, превращающий все ваши жесткие диски в одно логическое устройство. Если вам не хватает места для сохранения данных (а кому его хватает?), то интеграция новых дисков в стабильную инсталляцию – непростая задача; а использование LVM позволит динамически перераспределять свободное место между всеми вашими дисками более эффективно. Но сделать это можно только перед установкой системы.

Раздуть тома

LVM – хороший пример того, что вам может захотеться изучить при новой установке. Но существует масса других причин, будь то серьезная «чистка» системы или просто решение попробовать нечто новое – и Linux предоставляет больший контроль над текущими данными, чем любая другая ОС. По сравнению с OS X или Windows, Linux позволяет относительно легко производить миграцию текущих параметров конфигурации и данных на новую инсталляцию системы. Если вам повезет, вы даже сумеете оставить нетронутым ваш домашней раздел [**/home**], просто перезаписав корневой раздел в новой ОС. Без регулярной «чистки» все современные компьютеры подвержены потерям производительности и стабильности, и мы должны считать себя счастливчиками, которым не нужно начинать работу Linux с нуля. »

Шаг за шагом. Создание домашнего раздела

Когда вы будете переустанавливать Linux, дистрибутив захочет переразметить ваш жесткий диск. Идея хорошая, поскольку она гарантирует, что ничего не выживет, но это также означает отсутствие убежища для ваших данных. Создание отдельного раздела для данных, или домашнего – вот средство, которое позволит безопасно переустановить ваш дистрибутив без потери данных. Это самое удобное решение для тех, кто жаждет новых дистрибутивов.

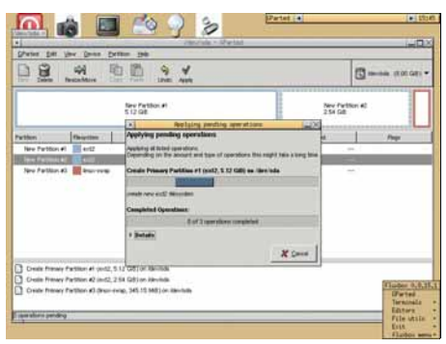


1 Запускаем GParted

Большинство дистрибутивов позволят вам переразметить диск в процессе инсталляции, используя инструмент типа *GParted*. Просто выберите диск, который вы хотите разметить, и удалите все существующие разделы. Это действие полностью уничтожит ваши данные, поэтому либо используйте новый диск, либо сделайте резервную копию данных, достойных сохранения.

2 Создаем разделы

Вам понадобятся три раздела: корневой, для домашнего каталога или данных, и для виртуальной памяти, или подкачки [swap]. Первые два раздела должны использовать файловую систему ext3 [не обязательно: можно также установить ReiserFS и ряд других, – прим. пер.], для swap-раздела нужен тип файловой системы «linux swap». Размеры первых двух разделов зависят от имеющегося свободного места, а размер swap-раздела должен в два раза превышать размер установленной оперативной памяти.



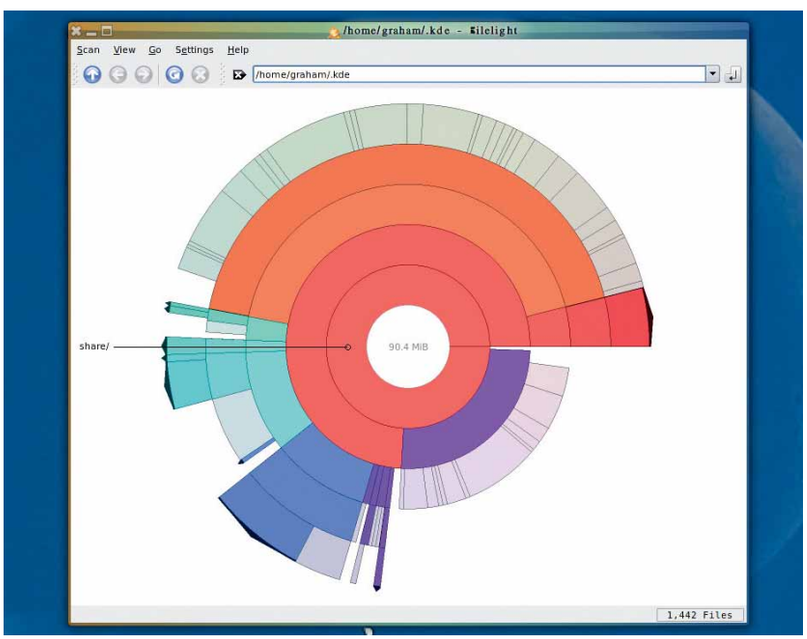
3 Применяем изменения

После нажатия кнопки Apply (Применить) все ваши изменения будут записаны на диск. После этого вы сможете выбрать эти разделы при инсталяции Linux. Не забудьте выбрать соответствующие точки монтирования для каждого отдельного раздела.

Подготовка и резервное копирование

Что вам нужно взять, куда это положить и как эффективно запланировать следующую установку.

» Все было бы относительно просто, если бы мы собирались установить Linux заново. Все мы неоднократно устанавливали и переустанавливали Linux, и самое главное при переустановке – перенести в новую инсталляцию важные данные и выбросить мусор, который тянет вашу систему назад. В принципе, это вроде несложно: скопировать содержимое вашей домашней директории на DVD, компакт-диск или USB-брелок, стереть текущую инсталляцию и установить систему заново, затем скопировать старую домашнюю директорию назад. Но так бывает редко. Во-первых, старые данные нужно почистить, во-вторых, вам, возможно, потребуется сохранить данные, находящиеся вне вашей домашней директории.



› Инструменты, подобные *Filelight*, помогут точно увидеть, где хранятся ваши данные.

Выбор программ

Все сводится к тому, какое программное обеспечение вы используете, и как приложения отличают ваши важные данные от собственных файлов конфигурации. Вы можете возразить: «Ну, большинство приложений позволяют мне сохранить мои документы в отдельную директорию» – а как же ваши закладки в браузере или тщательно отредактированый фон рабочего стола, выбранные шрифты, настройки и контакт-лист вашей программы обмена мгновенными сообщениями, ваша почтовая учетная запись и сама электронная почта? Большинство рабочих столов Linux держат эту информацию подальше от чужих глаз. Выигрывает стабильность (данные укрыты от воздействия ошибок пользователя), но миграция на новую систему превращается в кошмар.

Основная проблема в том, что переустановка редко сводится к повторному развертыванию той же самой версии дистрибутива. Большинство пользователей заодно обновляет свой Linux до последней версии, в результате чего новая инсталляция привносит новые версии ваших любимых приложений – к примеру, свежие релизы Gnome или KDE; а новые версии норовят менять файлы конфигурации. Если программист вашего любимого приложения был ленив, то ваши старые настройки теряют всякий смысл. Если программист халатно отнесся к работе, обновление может вызвать всевозможные непредсказуемые проблемы.

Например, программа обмена сообщениями KDE на вид может сперва работать как обычно, но вы очень быстро обнаружите, что не можете подключиться к сети MSN. Даже после удаления регистрационной информации и воссоздания учетной записи MSN по-прежнему работать не будет. Файл конфигурации поврежден, так как в настоящее время он содержит значения сразу и от предыдущих, и от текущих версий клиента мгновенных сообщений. Единственный вариант – это удалить файл конфигурации и повторить попытку. Именно поэтому, в большинстве случаев, простое обновление дистрибутива не срабатывает, и ручной отбор файлов конфигурации лучше, чем произвольное копирование всех найденных файлов.

Рассчитаем… рабочий стол

Существует несколько различных категорий приложений, которые мы должны учитывать. Первое и самое трудное – это рабочий стол: трудное потому, что он обычно представляет паутину приложений, разработанных толпой разработчиков, каждый из которых имеет свою идею о хранении настроек. Они могут уживаться на одном рабочем столе,

Что сохранять

Вопрос, какие данные сохранить, а какие отбросить, является жизненно важным при переустановке. Вы хотите сохранить все, что может быть полезно, но не ценой повторения проблем, которые и привели систему к «болезни». За начало с нуля аргументов немало, но есть много данных, которые можно спокойно сохранить, не опасаясь последствий. Вот список наиболее важных из них.

KDE

KDE – наихудший пример хранения полезной информации наравне с временным хламом. В идеале, вы можете перенести настройки в новую систему путем простого копирования скрытой директории **.kde** из старого раздела **/home** в новый, но на практике она будет содержать множество ненужных данных. Сосредоточьтесь на двух директориях: **~/.kde/share/apps** и **~/.kde/share/config**. Для каждого используемого вами приложения, вероятнее всего, есть файл конфигурации в директории **config**, а любые данные, связанные с приложением, можно найти в **apps**. Например, **Kopete** хранит список контактов директории **apps/kopete** и файл с учетными записями в **config/kopete**. Для сохранения настроек нужно скопировать обе директории. Точно также *Konqeuror* хранит закладки в **apps/konqueror/bookmarks.xml**, а настройки в **config/konquerorrc**.

Стоит сохранить данные и других приложений, включая *KMail* для настроек электронной почты, *KHotKeys* – для сохранения раскладок клавиатуры, *KWallet* для паролей и *Amarok* для вашей музыкальной коллекции. Вы можете предполагать, что ваши почтовые сообщения сохраняются в **.mail** или **.Mail** в вашей домашней директории, но это зависит от используемого почтового клиента. Например, с *KMail* вы должны сохранить директорию **share/apps/kmail/mail**.

Gnome

Внутренняя конфигурация Gnome более последовательна, чем в KDE, благодаря использованию базы конфигурации **gconf**. Вы, наверное, сталкивались с **gconf** при использовании **gconf-editor** для изменения различных настроек Gnome. В этой базе

но редко достигают соглашения о месте хранения файла конфигурации и способе им пользоваться. Многие из нас, столкнувшись с этим, норовят выкинуть все настройки и начать все с нуля, но это не всегда необходимо. При небольшой тщательной «расчистке» можно перенести практически все полезное в новую инсталляцию, главное – знать, что сохранять.

Каждый рабочий стол Linux придерживается своего подхода к хранению конфигурации, причем в KDE он наиболее унифицирован, но в любом случае готовьтесь работать со скрытыми директориями. Это оплот любых файлов конфигураций, и работа с ними – настоящая головная боль. Обычно для файлов конфигурации хорошо срабатывает метод «с глаз долой – из сердца вон», так как их редко приходится менять вручную. Но он явно не годится при резервном копировании данных, и вы должны точно знать, где они живут, чтобы чего-то не упустить при этом процессе.

Скрытые файлы и директории начинаются с точки '.' и обычно не видны. Чтоб их увидеть, в файл-менеджере необходимо включить показ скрытых файлов и директорий. В *Nautilus* из Gnome для этого нужно нажать Ctrl+H, а в *Konqueror* – выбрать опцию Показывать скрытые файлы из меню Вид. Причина важности показа скрытых файлов и директорий в том, что необходимо точно видеть, что происходит при «обрезке» различных файлов и директорий в их скрытой иерархии.

Если вы используете KDE, загляните в **.kde**. На этом уровне нас интересует только директория share. KDE подражает корневой файловой системе в хранении пользовательских настроек, и директория **share** – эквивалент **/usr/share** или той директории, которую ваш дистрибутив использует для сокрытия файлов KDE. Приложения KDE сохраняют свои настройки где-то в дебрях каталога share.

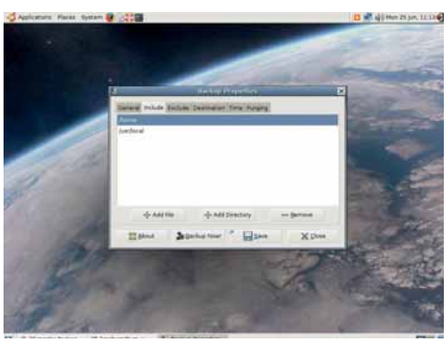
В Gnome ситуация совершенно иная, поскольку он использует базу данных gconf для хранения большинства настроек. Они находятся внутри директории **.gconfd** в виде XML-файлов, но вы опять-таки должны быть очень осторожны. Например, почтовый клиент *Evolution* использует gconf для хранения настроек вашей учетной записи, а директорию **.evolution** – для сохранения вашей почты. Если вы не уверены, что точно знаете, где найти информацию, но знаете, что она точно есть в вашей системе, расспросите других пользователей в Интернет, прежде чем нажать финальную кнопку Reformat.

Шаг за шагом: Резервное копирование

Путей выполнения резервного копирования ваших данных не меньше, чем способов сварить яйцо, но выбор самого легкого из них снижает шансы допустить ошибку. Мы используем программу *Simple Backup Suit*, или *sbackup*, но принципы сохранения/восстановления идентичны для большинства инструментов: создается архив с резервируемыми файлами, сохраняются их права доступа, и архив помещается в безопасное место для восстановления данных после окончания новой установки.

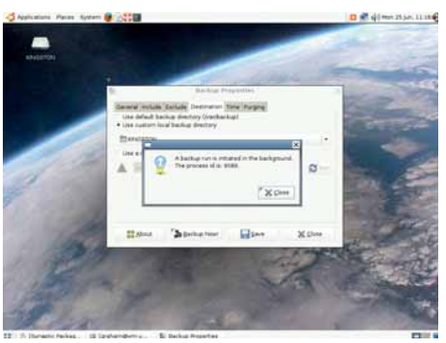
1 Устанавливаем инструмент

Установите *Simple Backup Suit*, используя ваш менеджер пакетов. Этот отличный инструмент с графический интерфейсом отлично справляется и с резервным копированием ваших файлов, и с выбором тех, которые необходимо восстановить. После установки его нужно запустить с привилегиями администратора.



2 Настраиваем, что копировать

Это самый важный шаг. Убедитесь, что вы учли все директории, содержащие ваши данные – например, домашнюю директорию **/home**; в ней могут быть вложенные директории с данными. См. врезку *«Что сохранять»* на стр. 28 для определения резервируемых данных, а если сомневаетесь в чем-то, лучше проявить излишнюю осторожность.



3 Сохраняем на внешнем носителе

Обязательно сохраните резервную копию данных на внешний носитель: диск или USB-брелок, сетевой диск, CD или DVD – либо пишите сразу на устройство, либо скопируйте данные вручную из **/var/backup** по завершении процесса резервирования.

хранятся настройки для большинства приложений Gnome. Учетные записи электронной почты для почтового клиента *Evolution* можно найти в папке **apps>evolution>mail**. Этот путь отражает, как данные хранятся в вашей домашней директории.

Каждый файл конфигурации, в формате XML, находится в соответствующей папке в скрытой директории **./gconf**. Для учетных записей почты *Evolution* вам необходимо сохранить, а затем восстановить папку **apps/evolution/mail**, хотя проще сохранить все содержимое папки *Evolution*. Однако ваша текущая почта сохраняется в директории **~/.evolution/mail**, поэтому действуйте осторожно. Через gconf вы также можете сохранить настройки других приложений, включая *Ekiga*, *Rhythmbox*, *Tomboy* и *Nautilus*. К сожалению, не все приложения Gnome gconf-совместимы. Например, для сохранения данных мессенджера *Gaim* [уже *Pidgin*, – прим. пер.] вам нужно резервировать директорию **~/.gaim**.

Общие приложения

Приложения, не относящиеся ни к Gnome, ни к KDE, любят использовать для хранения данных собственные локальные директории – это обычно скрытые директории с именами, совпадающими с названиями программ. Например, *Firefox* использует скрытую директорию .mozilla, где находится директория *.Firefox*. Вам не стоит сохранять все его модули, так как это может вызвать проблемы при переустановке, но ваши закладки должны быть сохранены. Это можно сделать либо через окно Менеджера закладок, либо скопировав файл **bookmarks.html**, находящийся в директории профиля **.default**.

Старые почтовые клиенты используют скрытую директорию **~/.mail** или, по-простому, «mail», и вы можете просто скопировать ее на резервный носитель. Существуют также настройки всей системы в целом [они хранятся в **/etc**, – прим. пер.], которые также можно было бы рассмотреть, но если вы используете различные базы данных и специальные конфигурации, то, скорее всего, знаете, где они находятся и как выполнить их резервное копирование.

Восстановление ваших данных

Вы уже сделали резервную копию ваших данных и переустановили любимый дистрибутив. Теперь пришло время восстановить их.

» Восстановление данных кажется плевым делом – ведь файлы с резервными данными нужно просто скопировать на тоже самое место? А вот и нет. Существуют подводные камни, которых нужно остерегаться. Самая важная деталь при пересоздании ваших предыдущих настроек – это права доступа к файлам. Переписав резервную копию во временную папку на жестком диске, посмотрите права доступа файла и его владельца, при помощи файлового менеджера или набрав *ls -l* в командной строке. Даже если вы создадите файлы с точно такими же именами и поместите их совершенно туда же в вашей домашней директории, существует вероятность, что дистрибутив начнет жаловаться на права доступа, и ничего не будет работать.

Даже если вы не встречались с правами доступа ранее, ничего сложного в их понимании нет. Каждый файл или директория имеет разрешения, связанные с конкретным владельцем, объединенной группой или всей системой. Разрешения могут быть установлены на «доступ для чтения», «доступ для записи» или «доступ на исполнение», или любую комбинацию из трех составляющих. Это правило распространяется на всю систему и используется для всего, начиная с вашей почтовой директории и заканчивая звуковой картой при проигрывании звука. Например, в Ubuntu, посмотрев права доступа звуковой карты, вы получите следующее:

```
ls -l /dev/dsp
crw-rw---- 1 root audio 14, 3 2007-06-19 09:41 /dev/dsp
```

Это означает, что root является владельцем устройства **/dev/dsp**, которое также принадлежит группе ***audio***. Оба (root и audio) имеют права чтения и записи (rw), и для устройства это означает, что вы можете и отправлять данные на него (воспроизводить звук), и читать с него данные (записывать звук). Серия прочерков показывает, что к данным нет доступа «для всех»: для доступа к звуковой карте вам нужно быть или администратором, или членом группы ***audio***. Права доступа на ваших собственных файлах работают таким же образом. После установки новой системы осмотрите сохраненные файлы со старой установки. Вы можете увидеть что-то вроде этого:

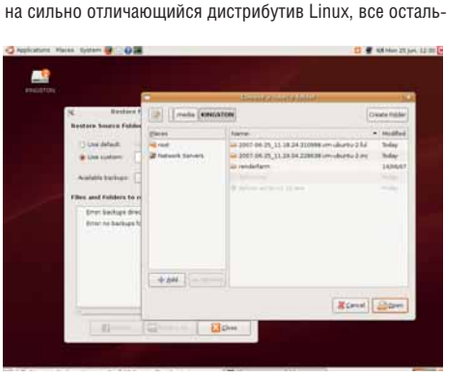
```
drwxr-xr-x 6 1002 1002 4096 2007-04-11 13:51 sakamoto
```

Где здесь имя владельца и группы? Проблема заключается в том, что в вашей системе имена служат лишь прикрытием для идентификационных номеров. Это похоже на работу Интернета, где имя сайта только маскирует его IP-адрес, и вашему компьютеру необходимо извлечь IP, прежде чем добиться какой-то пользы. Два значения номера 1002 представляют имя и группу владельца директории ‘sakamoto’, и причина, почему мы видим числа, а не имена, заключается в отсутствии таких пользователя и группы в системе. Вы можете обнаружить вместо 1002 два совершенно других имени пользователя и группу. Это объясняется тем, что другой пользователь [в новой системе] получил идентификатор (ID) 1002 и был подменен в выводе.

Ответом на все эти проблемы будет обеспечение таких же идентификаторов пользователей и групп, как и в старой инсталляции. Если вы поклонник одного и того же дистрибутива и создаете пользователей в

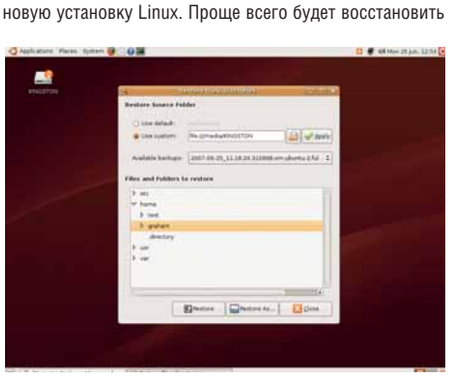
Шаг за шагом: Восстановление данных

В восстановлении хорошо то, что вы уже знаете, где все должно находиться. Директории Gnome и KDE будут на том же месте, и за исключением случая перепрыгивания на сильно отличающийся дистрибутив Linux, все остальные файлы надо поместить в их исходное положение. Мы будем использовать Simple Backup Suite для восстановления наших ранее зарезервированных данных на новую установку Linux. Проще всего будет восстановить нужные вам директории во временную папку и там отсечь файлы, которые вам больше не нужны.



1 Выбор резервной копии

Первый шаг – запуск *Simple Backup Suite*. Нужно точно указать место, где хранятся резервные копии. Если вы используете внешний диск, подключите его, затем выберите Use Custom и нажмите на значок папки, открывающей диалог указания пути к вашему устройству. Обычно оно находится в директориях **/mnt** или **/media**.



2 Восстановление на диск

После нажатия на Apply выведется список доступных резервных копий. Выберите из раскрывающегося списка вашу резервную копию. В нижней панели появятся файлы и директории вашей резервной копии. Выберите те, что необходимо восстановить, и нажмите кнопку Restore As для указания другого пути.

3 Чистка новых данных

Не стоит затирать текущие данные, пока вы не произвели чистку вашей резервной копии, поэтому сперва запишите ее во временную директорию. Потом по одному перемещайте файлы и директории в рабочую инсталляцию. В зависимости от типа восстанавливаемых фалов, может понадобиться перезагрузка компьютера для обновления системных настроек.

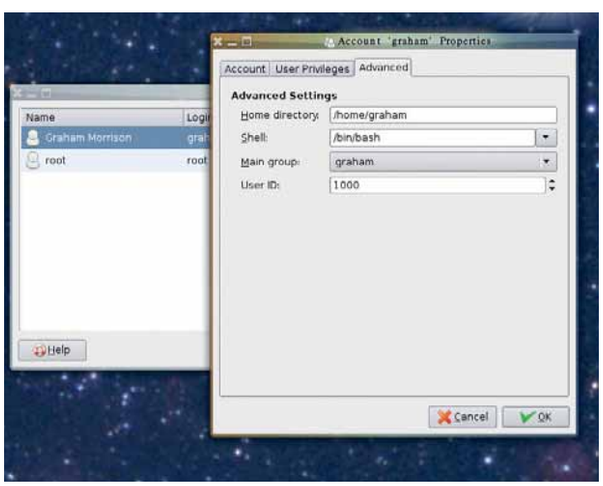
одинаковом порядке, то почти наверняка их идентификаторы будут совпадать. Но при смене дистрибутива можно нарваться на разногласие в значениях. Одни дистрибутивы нумеруют пользователей, начиная с 500, другие, как в нашем примере, с 1000. Большинство из них имеют в составе Менеджер пользователей, позволяющий вручную определить необходимые ID пользователей и групп. Но вы также можете использовать командную строку и команду «adduser». К счастью, о системных файлах конфигурации беспокоиться не нужно: они обычно принадлежат пользователю root, и эта область одинакова во всех дистрибутивах Linux. Root всегда имеет идентификатор пользователя и группы, равный 0, и все файлы и директории из резервной копии, принадлежащие root, будут восстановлены как его собственные.

Индивидуальный подход

Вы можете создать временного пользователя для восстановления ваших настроек; его можно употребить для проверки, действительно ли нужные вам приложения работают с восстановленными данными. Происходит ли соединение вашего менеджера сообщений? Уцелела ли ваша переписка? Если возникнут проблемы, вы сможете попробовать восстановить данные вновь, опустив часть не столь необходимых файлов и директорий. Даже минимум восстановленных данных все же лучше, чем ничего, получаемое при установке с нуля. Еще один плюс – вы узнаете гораздо больше о своей системе, и это поможет вам провести ее обновление в будущем и лучше понять идею использования отдельного домашнего раздела, что гораздо больше, чем вы получили бы, делая то же самое в альтернативной ОС.

Но для большинства из нас переустановка Linux раз за разом является частью нашей жизни. Вы, наверное, потеряли счет тому, сколько раз в течение многих лет вы вставляли в лоток привода загрузочный диск со «свежим» дистрибутивом. Мы все склонны к установке свежих релизов дистрибутивов и к уничтожению вполне рабочей системы кучей скачанного ПО – это вошло у нас в привычку. Так что у вас, несомненно, есть и свои соображения о том, как пережить шторм переустановки.

Даже здесь, в башнях LXF, мы все придерживаемся различных подходов к обновлению нашей любимой ОС. У каждого свои инструменты для сохранения данных в неприкосновенности – начиная со «слепого» обновления системы и заканчивая отделением домашнего раздела и вместительными USB-брелками. А вот Майк даже не помышляет об обновлении и продолжает использовать Red Hat 5.2 на своей Amiga. Вы можете просто списать свои персональные данные и файлы конфигурации или постоянно сохранять их на удаленном диске в вашей сети; можете просто нажать кнопку Обновить при новой установке системы; но что бы вы ни делали, мы с удовольствием послушали бы ваши советы по «выживанию при обновлении». Наверняка у вас имеется собственный взвешенный подход к решению проблемы, а мы хотим создать базу данных подобных решений – как для новичков, так и для экспертов. По-вашему, мы что-то упустили? Или у вас есть соображения, способные кому-нибудь по-настоящему помочь? Тогда «сбросьте» нам вашу идею, или, расхрабрившись, добавьте ее нам в Вики (**www.linuxformat.co.uk/wiki**). Вы поможете работе множества других людей, а мы, в свою очередь, если наберем достаточно советов, сможем опубликовать обновленную версию статьи с лучшими из них. LXF



› Присваивать номера идентификаторов пользователя и группы гораздо проще через графический интерфейс.

Пять блиц-советов по очистке

1 Настрой дистрибутив под себя
Revisor, поставляемый с последним релизом Fedora, идеален для создания собственного настроенного дистрибутива и помещения его на USB-брелок. Настроенный под себя дистрибутив дает вам возможность его тонкой настройки, а также сводит к минимуму хранимые данные.

2 Спасение не за горами
Если вы сталкиваетесь с проблемами при восстановлении из резервной копии или переустановке ОС, держите под рукой такие вещи, как *SystemRescueCd* для устранения неполадок. Он поможет как при проблемах с LVM и загрузчиком, так и при восстановлении резервных архивов и даже старых данных.

3 Клон рабочей установки
Использование *Partimage* позволяет сделать копию недавно установленного и настроенного Linux. Она поможет сэкономить много времени, если вам просто нужно быстро переустановить систему, а также послужит резервным источником.

4 Удаленное хранение
Храните ваши перемещаемые данные на другом диске. Тогда вы сможете стереть вашу систему, зная, что важная информация по-прежнему в безопасности. Если это хранилище находится в сети, у вас появляется добавочное преимущество всеобщего доступа.

5 Используйте символьные ссылки
Директории, связанные символьными ссылками, как вы знаете, можно получить с помощью команды *ln -s источник наЗначение*. Если пункт назначения находится на удаленном диске, то при новой установке достаточно будет обновить символьные ссылки, чтобы получить доступ к старым данным. Ссылки, что удобно, сохраняются при резервном копировании.

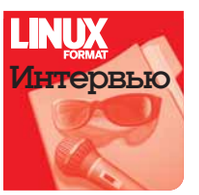
О СВОЕМ ЛИЧНОМ ВКЛАДЕ

«Чего я добился — так это попыток мыслить стратегически о том, что можно сделать, чтобы вовлечь в проект новых людей.»

Nick Veitch

GEGL идет в будущее

Студент, изучающий графику, и демо-художник в одном лице, **Ойвинд Колас** обосновался в самом сердце создания Грааля разработчиков GIMP – движке GEGL. Мы отправились в Монреаль, где он представлял обновление на конференции Libre Graphics…



У Ойвинда Коласа [Øyvind Kolås], более известного разработчикам *GIMP* и всем участникам этого проекта под ником «pippin», весьма непринужденные манеры. Я зажал его в угол с чашкой кофе, вытащив из зала, где проходила большая часть нынешней конференции Libre Graphics, и хочу поговорить с ним о GEGL: в этом вопросе он разбирается лучше всех в мире.

Проект GEGL стартовал больше семи лет назад. Его главной идеей было создание движка для обработки графики, способного решить все проблемы *GIMP*, на которые жалуются пользователи: скорость, отсутствие пространств цветов, ограниченная разрядность цвета. Однако изначальный проект забуксовал, едва появившись, и сдвинулся только тогда, когда в него заглянул Ойвинд Колас – из чистого любопытства. И теперь GEGL уверенно продвигается по прямой дороге к будущим версиям. Как же получилось, что Ойвинд Колас принял на себя полную ответственность за проект? **Ник Вейч** разберется.

Linux Format: Что привело вас к программированию инструментов для работы с графикой?

Ойвинд Колас: Я начал программировать всякую всячину, связанную с графикой, когда мне было лет 13–14. У меня есть опыт и в искусстве, и в компьютерах. После окончания школы я задумался, чем бы заняться дальше, и предпочел изобразительное искусство, решив стать дизайнером. Ну, и отправился в художественную школу, в Испанию. Я поступал в разные учреждения, и уже почти поступил, но не совсем: тут я вернулся к своему второму хобби и принялся за изучение мультимедиа и компьютеров, занимался этим года два, а потом вдруг стал преподавать!

Я работал над инструментами создания визуальных композиций. Так что я и кую эти инструменты, и сам же ими пользуюсь. Вначале я вышел на скандинавскую демосцену. Отсюда и началось мое увлечение компьютерной графикой.

LXF: Да, уметь совместить эти два хобби, должно быть, действительно круто. Как вы думаете, вы пришли бы к этому, если бы поступили в колледж искусств?

ОК: Я бы продолжал использовать компьютер – я и тогда уже писал свои небольшие инструменты.

»

LXF: Значит, это было неизбежно?
OK: Я всегда занимался и тем, и другим.

LXF: Давайте вернемся к настоящему. Вы — ведущий разработчик GEGL, движка, который, помимо прочего, станет сердцем будущих версий *GIMP*. И как дела у GEGL?
OK: Положение дел на данный момент таково: GEGL уже существует, и он делает то, для чего и был разработан. Пока что он не вполне оптимален, но у него есть небольшой публичный API, который – я уверен – надолго останется стабильным. Он достиг той стадии, когда создаются привязки для языков C#, Python и Ruby, и уже можно начинать рекомендовать его к использованию.

LXF: Я не заглядывал в код где-то месяц. Вы не введете меня быстренько в курс того, что происходит с последней версией?
OK: Последняя версия появилась как раз перед Fosdem. Несколько месяцев перед Fosdem у меня был прямо спринт, связанный с кодом, но с тех пор мне было както некогда с ним поработать. Разработка впала в спячку. В прошлом месяце у меня вылетела из головы вся кодовая база GEGL. А теперь я в процессе закидывания ее обратно.

LXF: По-вашему, это трудно? Трудно писать код для GEGL?
OK: Некоторые вещи в нем очень нетривиальны. Сейчас там есть масса интересных штук – сделать их нетрудно, но делать-то все-таки надо! То, что я сумел достичь той стадии, когда появился публичный API для Fosdem, было весьма утешающим. Теперь можно вернуться назад, перетрясти внутренности, увеличить скорость и заняться рефакторингом, не меняя внешний API. Все созданные скрипты будут и дальше работать, но в более поздних версиях скорость будет продолжать увеличиваться.

О ТЕКУЩЕМ СТАТУСЕ GEGL

«Уже созданы привязки для языков C#, Python и Ruby, и ими можно пользоваться.»

LXF: Это, должно быть, лучший способ – потому что постоянно видна осязаемая разница.
OK: Это я позаимствовал из другого проекта. Он фокусировался на стабильности API и весьма консервативно относился к любым изменениям в нем, но при этом добавлял в код инструменты, чтобы отслеживать изменения от версии до версии. Я тоже могу создавать графики процессов внутри кода GEGL и отслеживать, какие из них используют больше всего времени – скажем, обработка данных или преобразование пикселей.

LXF: Хотя в GEGL много участников, вам по большей части приходится работать над ним самому, не так ли?
OK: Да. Я не являюсь исходным автором, но…

LXF: А у вас нет ощущения, что вы предоставлены сами себе?
OK: Мне всегда помогали. Чего я добился – это попыток мыслить стратегически о том, что можно сделать, чтобы вовлечь в проект новых людей. Когда я начал воскрешать GEGL почти два года назад, ко мне обращались люди и спрашивали, чем они могут помочь, а я отвечал: «Не знаю». Мне потребовалось время на освоение существующей кодовой базы, на то, чтобы разобраться в ее работе, и на то, чтобы изменить то, что не нравилось другим или не нравилось мне.

Теперь, когда есть публичный API, этот API и есть GEGL. Сейчас простора для экспериментов куда больше, чем раньше. Все компоненты определены, и они работают.

Еще одна сфера, куда можно привлечь других – модули расширения. Это та часть API, которая пока что нестабильна, кроме кусочка, который отвечает за создание приложением конвейера для обработки изображения и настройки параметров для каждого действия. Но по части создания новых элементов обработки – тут такой гарантии нет. Причина в том, что начинка требует рефакторинга и реструктуризации, и я бы хотел добавить несколько функций, которых пока нет. И если мне удастся вдохновить кого-нибудь на написание модулей расширения, то нужно будет решить все эти проблемы.

LXF: Насколько высоко это все стоит в вашем списке приоритетов?
OK: Я еще не так далеко зашел.

LXF: Но ведь при каждом добавлении наверняка есть искушение взять да добавить еще что-то?
OK: Есть определенные вещи, которые пока невозможны, но должны стать возможными. И в процессе их превращения в возможные не исключено, что каждому модулю расширения придется изменять свое поведение. И поскольку я в курсе изменений, я не очень охотно публикую API. Он просто очень быстро устареет.

LXF: Раньше мы говорили о том, сколько времени вы сможете уделять GEGL в будущем. Можно ли считать, что сейчас он достиг такого состояния, что при необходимости эстафету может принять кто-то другой?
OK: М-мм... да.

LXF: Желающие найдутся?
OK: Я думаю, что заинтересованных найдется немало. Сейчас код в таком хорошем состоянии, что у любого получится поддерживать его в рабочем состоянии – и это не займет много времени. И теперь уже есть кое-какая документация, так что это в принципе будет возможно.

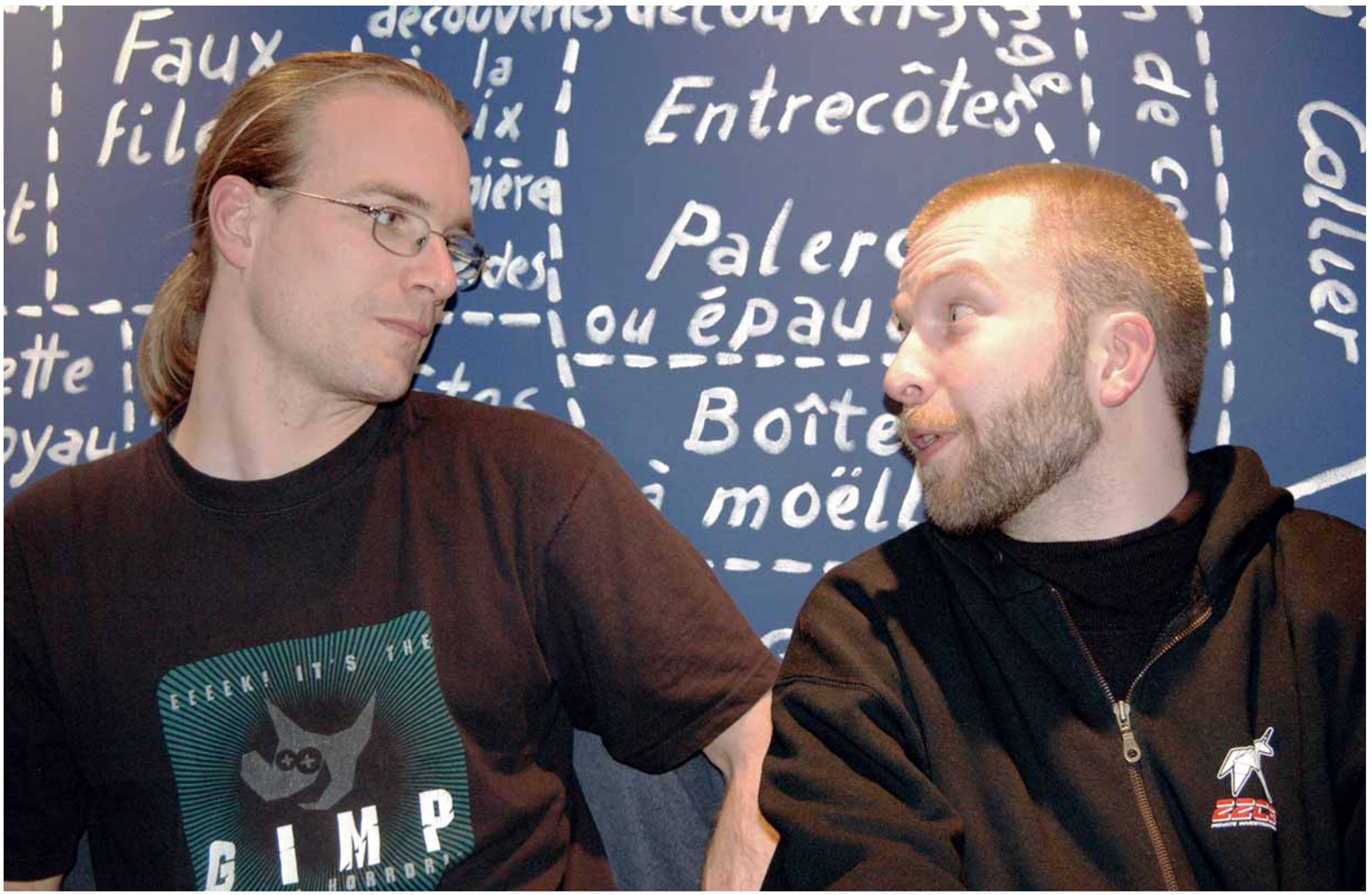
LXF: Намного проще, чем то, что делали вы…
OK: Я сильно боялся приниматься за GEGL. Когда я впервые загрузил код из CVS, я просмотрел его и ничего не понял. Мне очень понравилась сама идея. Поэтому я написал собственный прототип библиотеки обработки изображений, очень похожий на GEGL, а заодно использовал его для своего проекта редактирования видео. Процесс возрождения GEGL был фактически слиянием этих двух проектов. В моей программе обработки изображений уже имелись модули расширения, и она была частью полного приложения для редактирования видео, в котором можно было создавать композиции и добавлять спецэффекты. Мы с другом сделали несколько клипов с использованием этой программы.

LXF: Потрясающе, и они были хороши?
OK: Ну... один из них заработал какую-то награду в области независимого видео, а другой пару раз прокрутили на MTV. Так что это была работа по редактированию коммерческого видео. Опыт создания той библиотеки позволил вернуться к GEGL и подумать: «Так, ну и что должна делать эта программа?» Я осознал, что есть большие куски кода, которые не имеют ничего общего с остальными. То есть получалось три или четыре реализации одного и того же, часть из них не работала, и ни одна не стыковалась с центром.

Когда мы начали чистить код, этим занимались разработчики *GIMP*, Свен Нойман [Sven Neuman] и Митч Наттерер [Mitch Natterer], и они же следили за тем, чтобы все соответствовало основному направлению работы *GIMP*. А я в это время пытался втиснуть туда концепции, которые были в моем прототипе. Это было летом 2005 года.

Тогда много чего произошло, а потом какое-то время было затишье. И я создал библиотеку под названием *Babl*, которая была просто средой для преобразования изображений, вырванной из прототипа. У меня уже был код, но я отдельно выделил его для *Babl* а затем использовал её в GEGL. Для кода GEGL, сделанного в прошлом году, я многое позаимствовал из собственного кода, интегрируя в GEGL то, над чем работал последние пять или шесть лет.

LXF: Трудно будет бросить GEGL, если вдруг придется это сделать?
OK: Вообще-то я не собираюсь расставаться с этим проектом. Это моя игрушка. Мне самому подавай, чтоб была библиотека для работы с алгоритмами обработки



изображений. Еще я хочу попробовать несколько экспериментов по использованию GEGL.

LXF: Как движок он имеет множество потенциальных возможностей использования. Понятно, что в нем заинтересованы разработчики *GIMP*, ведь он обеспечит работу будущих версий этой программы. Но людям интересно: а годен ли он и для других приложений?

OK: Надеюсь, да. Поэтому я и занимаюсь и привязками, о которых говорил раньше, и стабильностью API. Это попытки вдохновить других использовать его и создать вокруг него солидное сообщество пользователей. Много чего мы никогда не узнаем, пока люди не начнут этим пользоваться. И также станет возможным иметь общие модули расширения.

LXF: Даже доступ к цветовым пространствам пригодится некоторым простым приложениям. Но пока что никакие другие крупные проекты не заинтересовались GEGL?

OK: Ну, привязки к C# делает человек, работавший в проекте F-spot, а привязки к Python задумывал один создатель анимационных программ. Много всего разного происходит.

LXF: Странно было бы, если бы охотников не нашлось. Отложим на время GEGL в сторону: есть ли другие проекты, которые вас интересуют?

OK: У меня не так уж мало проектов, которыми я занимаюсь, и в частности, меня очень интересует интерфейс пользователя и его дизайн. Я подхожу к пользовательскому интерфейсу экспериментально. Когда-нибудь я, возможно, напишу нечто вроде *Aperture* или другого приложения для работы с фото. Я экспериментировал со множеством техник. В данный момент у меня есть пара безумных идей, но я не уверен, что они сработают.

LXF: Вы никогда не ощущали, что на вас давят, потому что *GIMP* смотрит на GEGL как на движок для своих будущих версий?

OK: Ну, это, конечно, еще одна мотивация для меня. Но это не такое уж давление. GEGL находится в такой стадии, что как только они [разработчики *GIMP*] начнут им пользоваться и поинтересуются, почему некоторые функции тормозят, у них найдутся талантливые программисты, способные помочь с GEGL.

LXF: Полагаю, что деятельность начнется весьма кипучая – это будет отличной проверкой работы GEGL...

OK: Когда я рассказывал, что было сделано раньше, некоторые из упомянутых изменений относились к производительности и к тому, как архитектура справляется с требованиями производительности. С некоторыми вещами уже все ясно – по типу: допустим, сейчас скорость невелика, а вот изменим то-то и то-то…

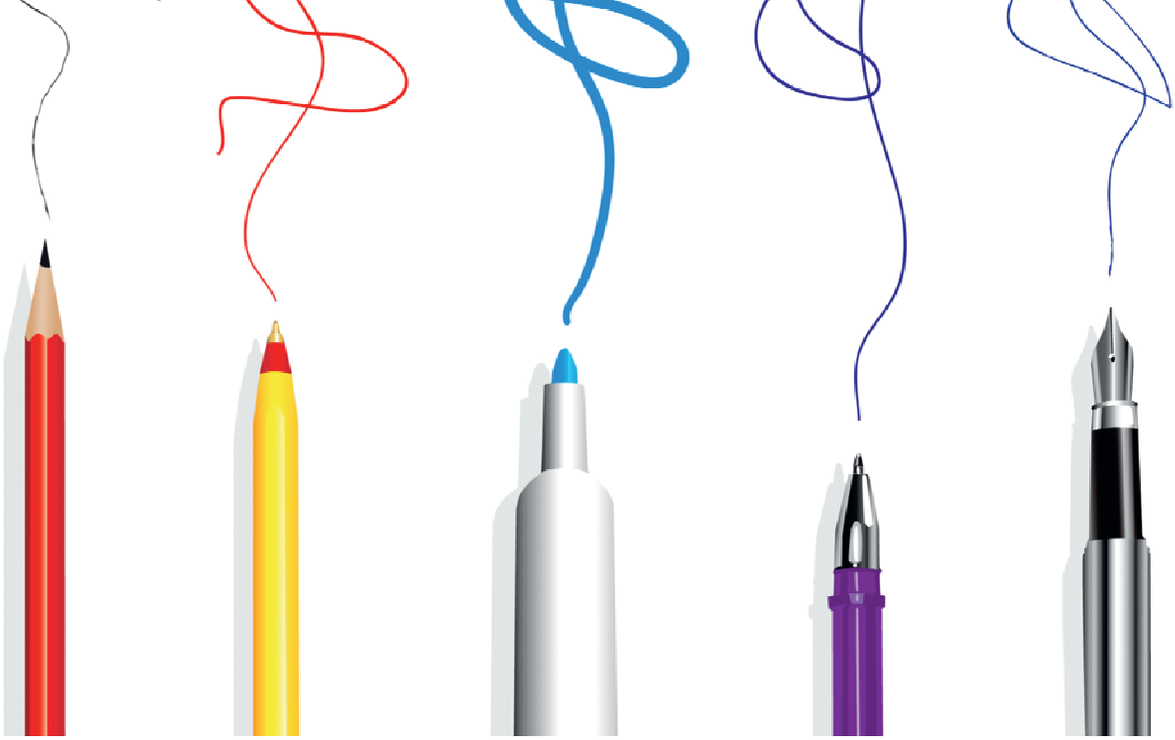
Однако это не причина их не интегрировать, потому что все эти изменения будет внутренними и не будут отображаться в публичном API.

LXF: И когда мы увидим первый GEGL, на котором работает *GIMP*?

OK: Когда он будет готов!

LXF: Я уж было решил, что в этом году, однако сейчас моя уверенность ослабела…

OK: Чтобы оптимизировать работу, я решил остаться с GEGL. Это важно и для меня, и для основных разработчиков *GIMP* – они знают, что кто-то работает над GEGL, и программа работает хорошо, и когда наступит время интегрировать их, есть человек, который знает, что и как надо сделать. В идеале нам бы хотелось, чтобы это произошло быстрее, но тут трудно судить. LXF



ПРОФЕССИОНАЛЬНАЯ ВЕРСТКА В SCRIBUS

По-вашему, Linux'у не по зубам издательское дело? Может, вы и правы, но **Ник Вейч** решил переубедить вас, создав эту статью при помощи только открытого программного обеспечения.

Издательская индустрия кардинальным образом изменилась с приходом в нее компьютеров, и в частности, с появлением Apple Macintosh. Первые Макинтоши имели прекрасную поддержку шрифтов, и не удивительно, что программное обеспечение для настольных издательских систем появилось сперва на них. Так, уже на заре использования компьютеров, профессионалы издательского дела оказались привязаны к оборудованию Apple. Вероятность того, что Linux придет в эту область и изменит все в одночасье, очень мала.

Но мы поговорим не о том, может ли Linux заменить собой Mac в сердце рынка DTP, а о его технических возможностях для выполнения таких работ. Можете ли вы использовать Linux для создания журналов должного качества, печатая их с использованием технологии «сразу на печатную форму (direct-to-plate)» на современных принтерах? Единственный способ выяснить это – проверка на практике.

В рамках статьи, мы немного умерим свои амбиции: вместо целого журнала попробуем сделать эти четыре страницы, используя только программное обеспечение с открытым исходным кодом. Они будут выведены в виде PDF-файлов и запущены в производственный цикл издательства Future Publishing так же, как если бы были сделаны обычным порядком (в Adobe InDesign на Mac). Чтобы проверка была честной, мы не сказали об этом никому вне команды LXF (если сейчас вы видите пустые страницы, значит, у нас ничего не получилось). Другое небольшое извинение состоит в том, что вместо профессиональной художественной обезьяны, верстающей страницы, я все сделаю сам, так что если они будут отходить от обычных стандартов, вы уж меня простите. Теперь – вперед!

Цветовые профили ICC

При конвертации изображений из одного формата в другой и получения результата, близкого к вами задуманному, хорошей идеей является использование цветовых профилей ICC. Помимо стандартных профилей от Adobe (см. «Вопрос изображений») существует множество других – например, для вашей цифровой камеры. Дополнительную информацию и программное обеспечение, связанное с ICC можно получить, пройдя по ссылке www.eci.org/eci/en/060_downloads.php. Кроме того, убедитесь, что вы включили управление цветом в меню настроек Scribus, если хотите использовать данный профиль для окончательного вывода.

Точка отсчета

Первым делом, еще до начала работы, нужно выяснить, как прийти к конечному результату. Обычно, в настольных издательских системах, программа верстки – это место, где различные объекты соединяются вместе, стилизуются, а затем облекаются в уместный дизайн. Используемые элементы зависят от конечной цели работы, но стоит отметить, что в большинстве документов встречается как минимум три типа содержимого.

Текст: Если вы не делаете чисто визуальное объявление с парой слов на нем, то этого материала у вас будет очень много. Эта та часть, которую люди читают.

Фотографии: Многие публикации содержат фотографии. Также вы можете использовать сгенерированные на компьютере реалистичные изображения, экранные снимки или что-то другое. Короче, имеется в виду растровая графика.

Иллюстрации: Даже если это небольшой элемент страницы, вроде логотипа или заголовка, или просто метки для совмещения страниц, вы, вероятнее всего, захотите использовать в своей публикации векторную графику.

Для создания каждого из этих объектов понадобится специальное приложение. Да, можно набрать текст прямо в *Scribus*, но это нежелательно. Встроенный редактор очень хорош для оформления текста различными стилями, но не так хорош для его написания, особенно если вы не можете жить без таких возможностей, как, например, счетчик слов.

При известной самоотверженности и умении, вы даже можете установить стили *OpenOfiice.org*, которые будут соответствовать стилям конечного документа – это может

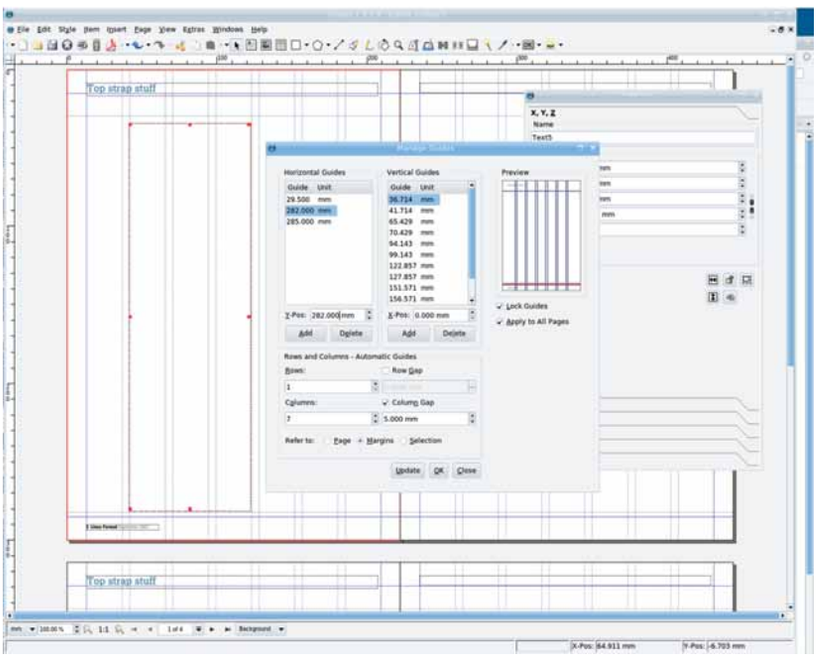
устранить подверженный ошибкам этап назначения стилей в тексте, но тогда придется доверить этот этап также и автору текста! В противном случае, очень легко назначить стили текста, используя встроенный текстовый редактор *Scribus*. Советы по стилям *OpenOffice* приведены во врезке «Забавы со шрифтами!» на стр. 39.

При желании можно, конечно, использовать старый текстовый редактор. Нет желания? Тогда двигаемся дальше.

Вопрос изображений

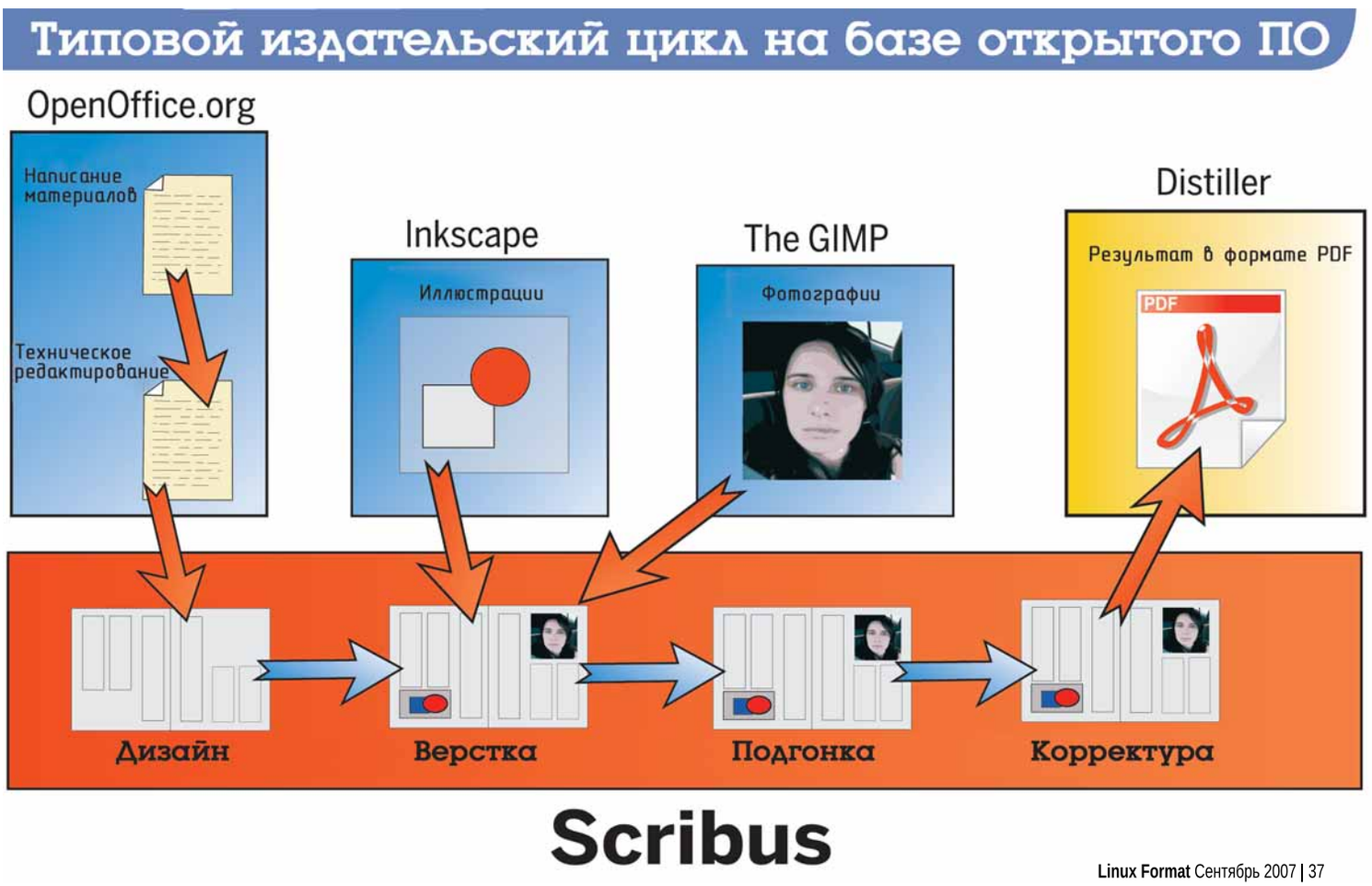
Казалось бы, по части обработки изображений *GIMP* – именно тот чудесный инструмент, что удовлетворит любые ваши нужды. А так ли это? Во многих случаях *GIMP* прекрасен. Можно изменить цвет, добавить лихие эффекты и вообще превратить любую картинку, экранный снимок или фото в то, что вам хочется. Однако в офсетной печати, применяемой при больших тиражах, на страницу наносят четыре краски: голубую (**C**yan), желтую (**Y**ellow), пурпурную (**M**agenta) и черную, она же основная (**K**ey). Поэтому для полноцветной печати (**CMYK**) необходимо иметь все изображения в формате CMYK. Все бы ничего, но для растровых файлов это создает проблему, так как GIMP не поддерживает цветовое пространство CMYK (поддержка CMYK появится при переходе на движок GEGL, где-то через год).

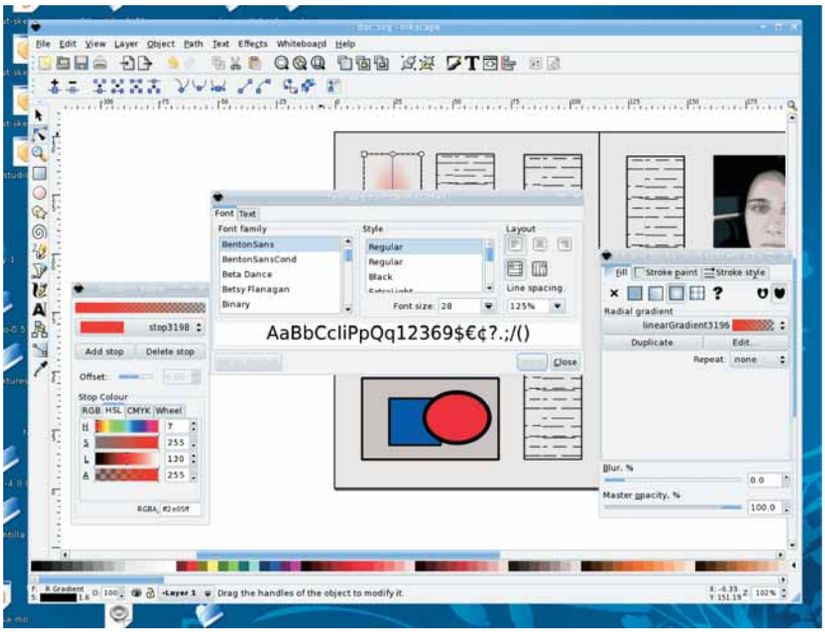
Есть два способа обойти эту проблему (на самом деле, способов масса, но лишь два из них заслуживают рассмотрения). Первый – воспользоваться модулем расширения *Separate* для *GIMP*. Он преобразует файл в формат CMYK. Раньше *Scribus* малость спотыкался на файлах, сгенерированных подобным образом, но начиная с версии 1.3.3.6 этот путь стал безопасен. Модуль *Separate* находится по адресу: www.blackfiveservices.co.uk/separate.shtml. Для его использования вы должны иметь установленные ICC-профили. Некоторые из них распространяются в архиве *Separate*, а файл **readme** содержит полную инструкцию по установке. Но чтобы сэкономить на головной боли, установите еще и ICC-профили от Adobe: они дадут вам лучший результат при выводе и преобразовании (www.adobe.com/support/downloads/). Для получения дополнительной информации, см. врезку «Цветовые профили ICC» на предыдущей странице.



› Правильное размещение сетки в проекте критично для любого дизайна и компоновки.

Второй путь – спокойно работать в *GIMP*, затем с помощью другого пакета отделить изображение и сохранить его в CMYK. К счастью, в Linux есть для этого инструмент





› ***Inkscape*** **бесценен для работы над иллюстрациями и небольшими «украшательствами».**

Cinepaint – давнее ответвление от *GIMP*, довольно элементарное, но включающее поддержку CMYK. Гораздо лучше, однако, *Krita* (часть пакета *KOffice*), которая также поддерживает преобразование из RGB в CMYK при различной разрядности цвета и использует встроенные профили. Хотя *Krita* не имеет таких возможностей для манипуляции с графикой, как *GIMP*, она обладает прекрасными способностями для решения большинства задач обработки изображений. Но вам все равно понадобятся профили изображения.

Что касается формата файлов – как правило, люди используют TIFF, хоть он и несколько чужд Unix-миру. Но ничто не мешает вам использовать формат JPEG (который также поддерживает CMYK). Для нашего производственного процесса мы будет выводить PDF-файлы с типографским разрешением, так что данный выбор некритичен – главное, чтобы полученный файл можно было встроить в PDF. Если вы сами и печатаете, то, очевидно, сгодится любой формат файла, и незачем беспокоиться насчет создания файлов CMYK (хотя это желательно, потому что дает на экране четкое представление о том, как будет выглядеть результат).

Сверка по вектору

С векторной графикой легче. Большинство векторных инструментов в Linux позволяют выбрать формат SVG (Scalable Vector Graphics). Это положительный момент, потому что *Scribus* хорошо его поддерживает. К отрицательным можно отнести необходимость остерегаться вашего возможного формата вывода. *Scribus* поддерживает SVG и множество его «хитростей», а вот PDF-формат их не поддерживает. Существуют некоторых вещи, в основном связанные с прозрачностью и другими эффектами, которые не будут работать в PDF-файлах, предназначенных для коммерческой печати, как в нашем случае.

Коротко говоря, если вам нужен экспорт в формате PDF 1.3, то нельзя применять прозрачность в векторной графике. Это относится не только к объектам с установленным основным прозрачным цветом, но также и к имеющим полупрозрачную заливку или градиент с прозрачностью. Это не проблема *Scribus* или *Inkscape*, а ограничения PDF-формата. Файлы в формате PDF версий 1.4 и выше поддерживают прозрачность, но могут быть совместимы не со всеми просмотрщиками и другими приложениями.

Некоторые эффекты *Inkscape* вообще нельзя использовать в *Scribus*, например, популярный эффект размытия (blur). Обычно *Scribus* предупреждает о возможной несовместимости при первоначальном импорте графики или во время префлайта – «проверки перед взлетом», как раз и предназначенной для выявления узких мест и потенциальных проблем. Если вам непременно нужно импортировать в *Scribus* изображения с такими эффектами, остается сохранить их как растровые изображения, правда, ценой возможной потери качества и удобства использования.

SVG-графика обрабатывается в *Scribus* с помощью модуля расширения, но не аналогично растровому изображению. Вместо этого она переводится в объект *Scribus* (тоже векторный), и впоследствии вы сможете ее легко редактировать.

Верстаем

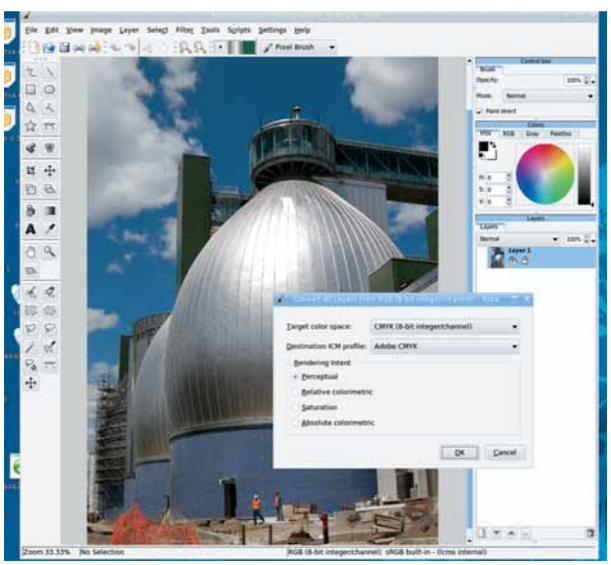
Все материалы подготовлены, и мы наконец готовы импортировать все в *Scribus* и создать страницу нечеловеческой красоты – или не готовы? Я – нет, и вы, вероятно, тоже. Перед тем как приступать к дизайну страницы, полезно обзавестись шаблоном, или, как говорят полиграфисты, «модульной сеткой». В случае с Linux Format нам очень повезло, так как он (она) у нас уже есть. Прекраснейшие в мире дизайнеры долгими месяцами трудились не покладая рук, чтобы придумать проект, объединяющий должную практичность журнала с изяществом и элегантностью. Затем в последний момент мы все это поломали, чтобы впихнуть побольше слов на страницу... Но так или иначе, в итоге мы имеем сетку для Linux Format и можем просто скопировать ее в документ Scribus, чтобы получить желаемый результат. Если вы разрабатываете собственную сетку, полезно проследить за этим процессом и разобраться, какие виды элементов мы использовали.

1. Размеры страницы. Основная часть дизайна – это физический размер страницы («обрезной формат» в полиграфических терминах). По ряду причин Linux Format использует не совсем равный A4 размер листа 222x300 мм плюс три миллиметра на выпуск (своего рода буфер вокруг пропечатываемой области).

2. Поля. Физические границы обычной печатной площади страницы. Они ведут себя как направляющие и легко устанавливаются в свойствах документа.

3. Направляющие. Linux Format использует довольно гибкую систему сетки, которая позволяет иметь две колонки для статей, четыре колонки для писем, три для обзоров и

› **Пройдя долгий путь, *Krita* стала полезнейшим инструментом преобразования в CMYK для печати.**



так далее. Некоторые журналы используют отдельную сетку для каждого раздела. В нашем везде используется сетка из семи колонок, которые применимы для всех вариантов (например, в двухколоночном тексте мы используем по три колонки сетки на колонку текста, и одну запасную для врезок или подписей).

4. Элементы. Используя пункт Правка > Мастер-страницы (Edit > Master Pages) из меню, мы можем добавить неизменяемые объекты, такие как верхние и нижние колонтитулы. Все, что помещено на мастере, появляется на каждой странице, но остается заблокировано.

5. Цвета. По умолчанию, *Scribus* предлагает палитру с поистине безграничными возможностями. Для большинства работ вы, вероятно, ограничитесь четырьмя или пятью цветами: в случае необходимости вы всегда сможете добавить еще, а при списке из 200 цветов трудно упомнить те, которые уже использовались.

6. Стили. Вы можете импортировать стили документа из текстового редактора, но, вероятно, вам по-прежнему будут нужны стили, которые есть только в *Scribus* (например, номеров страниц, колонтитулов и т.д.). В конечном итоге оказывается проще описать все стили в *Scribus* (вы сперва можете импортировать их из документа, если хотите).

Создание этой статьи!

Создание статьи о создании этой статьи в *Scribus* оказалось весьма непростым занятием. Не потому, что случались сбои в работе инструментов, а потому, что довольно сложно понять, с чего начать, не превратив статью в ссылающуюся-самое-на-себя ерунду типа змея Ороборос [Змей, кусающий свой хвост, символ бесконечности].

Поскребя в затылке, я прикинул, что самое лучшее решение – написать как можно больше текста, а уж потом как следует взяться за верстку. Поэтому я начал с установки сетки – приняв сетку типичной статьи LXF за основу.

В этом месте я прошу прощения у нашего художественного редактора Эффи и всех читателей в том, что часть из кусочков и кусков не совсем верны. Некоторые вещи было почти невозможно в точности воспроизвести в *Scribus* – например, скругленные углы плашек для заголовков врезок: они были первоначально сделаны вручную в *InDesign*, и мне пришлось делать их вручную в *Scribus* (двойной щелчок по любому элементу для редактирования узлов, и вы тоже можете играть с контрольными точками кривой Безье). *Scribus* тут не виноват, и я думаю, что если бы мы переделали дизайн журнала под *Scribus*, это проблема отпала бы сама собой.

Основная проблема, которая возникала в процессе работы, заключалась в том, что функция Undo (Отменить) в *Scribus* немного ненадежна – иногда работает, а иногда нет. Имелись также отдельные проблемы с графикой, особенно после того, как мы использовали функцию «collect for output» (собрать для вывода) и не вся графика скопировалась в папку назначения (это дефект *Scribus*, или просто сбой в моей системе?) .

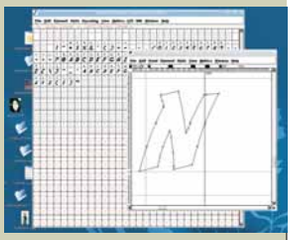
[**Внимание!** Следующая пара абзацев и наш собственный опыт вынудили нас написать собственный – и вовсе не в двух словах – комментарий к статье. Мы в российской редакции столкнулись не только и не столько с этими проблемами, к тому же не могли упустить такую возможность предупредить наших читателей о некоторых важных особенностях текущих релизов *Scribus*. Читайте наш чуть более скептический комментарий на следующей странице, – прим. росс. ред.]

Главным сюрпризом оказалась возможность сделать все. Кроме потери двух дней при попытке заставить работать шрифты (опять же не по вине *Scribus*)! В статье мы не выделили курсивом названия [это касается английской версии – прим. росс. ред.], потому что шрифт BentonSans (какой идиот его выбрал?!), не имеет курсивного начертания. В *InDesign* это кое-как решалось искажением текста, что в *Scribus* невозможно (только в нестабильной версии 1.4!) .

Однако к концу процесса я обрел практически полную уверенность в том, что *Scribus* обеспечит прочную и надежную работу. Согласно нашему технологическому процессу здесь в Future Publishing, сгенерированные *Scribus* PDF-файлы были еще и скормлены нашей собственной препресс-системе (которая генерирует PDF-файл, содержащий сгенерированные PDF), а это добавочная сложность, о которой большинству людей беспокоится не надо.

Вы сами можете судить об успехе этого эксперимента. Я гарантирую вам, что эти четыре страницы были получены на 100% свободном ПО, работающем под Linux, а не так уж много журналов могут этим похвастаться! Это было даже не трудно, хотя прежде чем вы начнете массово писать, я все-таки не верстаю журнал каждый месяц! **LXF**

Забавы со шрифтами



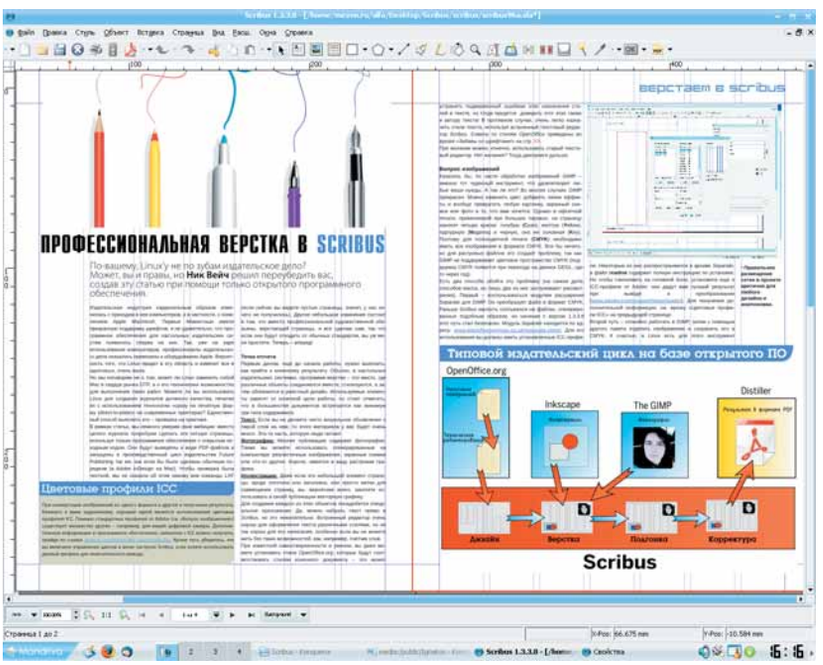
Вы можете столкнуться с той же проблемой, что и мы – потребностью использовать шрифты с вашего Макинтоша на машине с Linux. С PC будет не так уж много проблем, или, если быть честными, не будет проблем со шрифтами TrueType, которые можно просто скопировать (вы должны убедиться в том, что лицензионные ограничения позволяют копировать шрифт на другие системы – многие лицензии на шрифты этого не разрешают).

Шрифты PostScript Type1 все еще широко используются на платформах Mac, и это может вызвать большие проблемы. Они часто упакованы в Mac'овский «Портфель шрифтов» (Font SuitCase), но даже если этого нет, они подвержены любимой забаве Mac'а с разделением данных и ресурсов (шрифт Type1 состоит из нескольких файлов и, скопировав шрифт, вы обнаруживаете его пустым, поскольку все данные о шрифте содержались в другой, скрытой части).

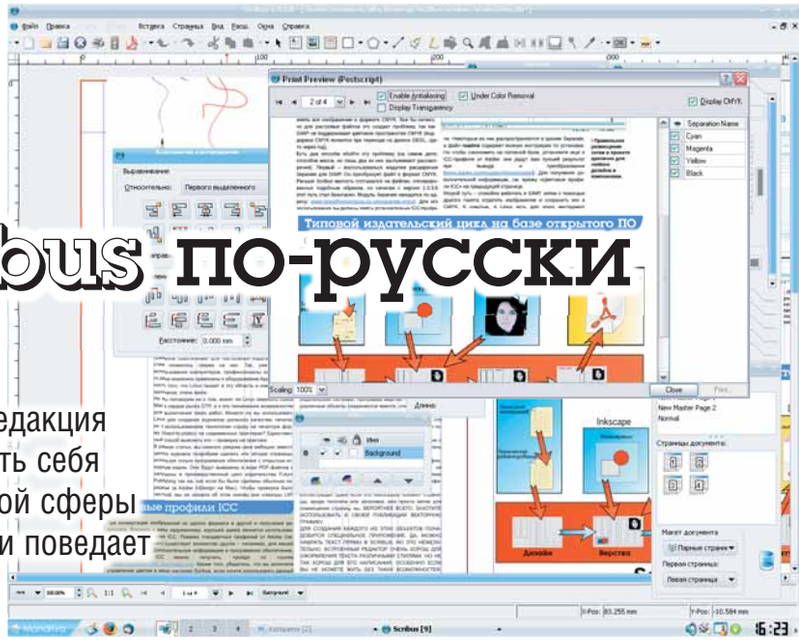
Решение этой проблемы (мы обнаружили его, провозившись немало часов) – это загрузить программу для Mac с открытым исходным кодом *fondu* (http://fondu.sourceforge.net/). Используйте ее для извлечения PFB-файлов, которые понимает Unix. Затем поместите их в ZIP-архив и скопируйте.

Под Unix легче всего загрузить PFB-файлы в *Fontforge* и сохранить затем в формате TrueType перед установкой шрифтов в вашей системе.

ВНИМАНИЕ! Читайте комментарий русской редакции, касающийся работы с кириллицей.



Верстаем в Scribus по-русски



Быть первопроходцем всегда трудно. Быть вторым ненамного легче, но куда обиднее. И российская редакция LXF решила, что у нее достаточно оснований считать себя «практически первой» в деле перевода издательской сферы на рельсы Linux. О «трудностях перевода» как раз и поведает **Родион Водейко**.

Наш эксперт

Родион Водейко всю сознательную IT-жизнь провел, сидя под окнами. Последние два года отвечает за выпуск нашего журнала и надеется перейти на полный Open Source. Готовый активист общества «Анонимных виндузятников» – оному осталось только появиться на свет...

Проще говоря...

Разницу в подходах программ к верстке можно легко объяснить и программисту: есть языки, требующие определения переменных в самом начале, перед тем, как вы начнете оперировать ими, а есть другие, позволяющие задавать их по мере необходимости. Продукты Adobe – это BASIC верстки. *QuarkXpress* и *Scribus* – ее ANSI C.

С тех пор, как мы получили этот номер и осознали совершенный нашими английскими коллегами подвиг (а как еще назвать сверстанный средствами Open Source материал?), нами двигало желание не уронить планку. Мы должны были повторить это достижение, попутно улучшив его, ведь нам предстояло соорудить статью в *Scribus* на русском, т.е. испытать все прелести локализации.

Наши опасения в целом подтвердились, зато мы приобрели бесценный опыт создания абсолютно свободных (во всех смыслах) публикаций, подняв уровень свободы самовыражения на невозможную в проприетарном мире высоту. И мы просто обязаны им поделиться.

Однако, вынужден сразу несколько разочаровать прожженых линуксоидов – комментировать буду я, ответственный за компьютерную сборку Linux Format, старый работник издательской нивы и при этом прожженный «форточник». Это был мой первый опыт серьезной и сразу – профильной работы в Linux, а потому не ждите триллера в стиле «Как я хакнул *Scribus*». Я не разбирался с премудростями свободной ОС, а лишь сделал свою работу в поставленных мне условиях. Потому мой рассказ будет о *Scribus* как таковом – с точки зрения обычного пользователя подобного программного продукта.

Итак, в нашем распоряжении были:

- *Scribus 1.3.3.8* и Mandriva Linux 2007.1
- *Scribus 1.3.3.9 portable* под Windows XP – версия, достаточно давно установленная мной на привычном рабочем месте, тоже использовалась в работе – отчасти для сравнения, отчасти – вынужденно.

Но – обо все по порядку. Как известно, в мире верстки два «супергероя» – компании Adobe и Quark. У каждой из них – по «суперпродукту», и к обоим прилагается своя рабочая концепция, даже идеология. Потому и верстальщики делятся на две большие группы (не без взаимопроникновения между ними, конечно) – каждая привыкла к своей программе и ее концепции верстки. Вкратце, разница такова: *QuarkXpress* предполагает полную разметку документа с дальнейшим заполнением ячеек подготовленным вовне содержимым. Иными словами, чтобы вставить в публикацию текст, вам придется нарисовать текстовые фреймы, а вставка изображения потребует наличия фрейма графического. *Adobe InDesign* (как ранее и *PageMaker*) допускает и такой подход, и прямую вставку текстового и графического содержимого – при этом необходимые фреймы все равно создаются, но попутно и (полу-)автоматически.

Scribus – однозначно в лагере *Quark*. И мне, мигранту с *InDesign*, это создавало дополнительные трудности в и без того нелегком процессе монтирования страниц на совершенно новой и малоизученной платформе. Сейчас я отделю зерна от плевел в своей голове (ну хотя бы попытаюсь) и расскажу вам о тех лишь узких местах, что существуют в *Scribus* для любого верстальщика вне зависимости от его привычек.

Резюме – эту военную тайну – выдам сразу: на сегодняшний день *Scribus* явно сыроват, несмотря на то, что работа над ним ведется колоссальная, и ее результаты видны в каждом новом микрорелизе, отмечаемом изменением аж четвертой цифры в номере версии! В первые полчаса работы я испытывал странное дежавю: точно так же 12 лет назад я впервые сидел за *PageMaker* и точно так же не знал, куда ткнуться, и что за этим последует. Но тогда у меня не было за плечами этих самых 12 лет опыта верстки в самых разных программах…

Ник Вейч, однако, нисколько не лукавит, когда радуется тому, как все прошло – *Scribus* действительно вполне рабочий инструмент в том смысле, что 95% всех возможных издательских работ выполнить в нем уже можно. Это само по себе – огромное достижение Open Source. На сегодня главный вопрос в ином – какую цену вы готовы заплатить за отсутствие ценника на *Scribus*? Парадокс, но время – деньги, а *Scribus* довольно медлителен. Каждый верстальщик со стажем знает неповоротливость «Индюка» (*Adobe InDesign*), однако, поверьте: *Scribus* умудряется затмить этого проприетарного монстра – причем в элементарных, самых нужных, рутинных операциях. Прежде всего, это касается редактирования текста.

Не верьте тому, кто скажет – мол, текст надо готовить в текстовом процессоре, а DTP – это только для верстки. В теории это верно, и *Scribus* отлично втягивает заранее отформатированный текст, однако в периодических изданиях, как ни крути, приходится постоянно править и форматировать текст прямо на полосе.

И вот тут наступает первый критический момент. Причем длится он с начала процесса вставки текста в публикацию и до полного окончания работы с этим текстом. А это очень долго! Ибо *Scribus*:

1. Понимает далеко не все принятые форматы текстовых файлов. 1.3.3.8 (тот, что под Linux – дальше мы будем говорить в основном про него, изредка сравнивая с Windows-версией) отказывался принимать что-либо, кроме SXW – родного формата *OOo Writer 1.x*. Возможно, это проблема конкретной сборки, но факт остается фактом. 1.3.3.9 под Windows (и это еще один парадокс!) куда менее капризен и принимает почти все, но файлы DOC теряют форматирование, в то время как документы *OOo* (SXW и ODT) – сохраняют разметку от и до. Главное разочарование – *Scribus* совершенно не воспринимает RTF, своеобразный Windows-стандарт обмена форматированным текстом, не привязанным к конкретному редактору (да-да, я про *Word*).
2. Видит далеко не все шрифты, несмотря даже на принудительное указание путей к папкам с ними. Закономерности в игнорировании некоторых гарнитур выявить так и не удалось.
3. Крайне неудобен для изменения параметров готового текста. Это, на удивление, вообще оказалось ахиллесовой пятой пакета. Для профессиональной издательской системы у *Scribus* совершенно негодный арсенал средств верстки основного (body) текста. Доступ к базовым свойствам абзаца (интерлиньяж, табуляции и отступы) очень ограничен. Все их разом можно настроить только при создании стиля абзаца. А символьные стили отсутствуют напрочь, что заметно выделение текста цветом, начертанием и с помощью иных атрибутов. Разные начертания шрифта (**bold**, *italic*) доступны только путем выбора другой гарнитуры, ибо каждое начертание – а это отдельный файл в системе – здесь считается (опять же, по Quark'овой традиции) отдельным шрифтом. Идеологически это, может, и правильно, но крайне неудобно. Судите сами: чтобы выделить текст *курсивом*, придется – вместо нажатия горячих клавиш или щелканья по кнопке в контрольной панели – открывать выпадающий список шрифтов и выбирать нужный (отметим, что в *Scribus* есть два пути редактирования текста, и если во встроенном текстовом редакторе все гарнитуры – читай, файлы – находятся в одном списке, то в панели свойств система двухуровневая: в одном списке выбирается шрифт, а в другом

Сеть в помощь!

http://en.wikipedia.org/wiki/Scribus – Вики о Scribus
http://wiki.scribus.net/index.php/Main_Page – справочная информация
http://web2.altmuehlnet.de/fschmid/docs/index.html – подробный учебник
https://www.redhat.com/promo/fonts/ – страница загрузки Red Hat Liberation Fonts, использованных нами

его начертание. Так или иначе, без утомительного «копания» в списках обойтись не удастся).

4 Текст, занимающий более одного фрейма (например, многостраничный) неудобно как создавать, так и редактировать. Во-первых, если вы не скомандовали *Scribus* «наваять» связанные текстовые фреймы при создании нового документа, то вам придется рисовать их самому, а потом вручную же связывать один с другим – сравните с функцией авторазмещения текста в продуктах Adobe!

Когда же текст все-таки разлит по полосам, боже упаси вас пытаться редактировать его весь скопом – вы ни за что не угадаете, какие из атрибутов распространятся на весь предусмотрительно (ха-ха!) выделенный вами текст, а какие придется применять к каждому фрейму отдельно. Я вам чуть-чуть помогу: выключка текста у меня сработала для всей статьи, а вот кегль шрифта и интерлиньяж (leading) пришлось менять кусками. С учетом некоторых логичных, но идиотских особенностей этого процесса, он вряд ли вызовет у вас желание когда-либо повторить его еще раз. Как ни странно, в Windows эта проблема стоит менее остро (или же это следствие перехода к версии 1.3.3.9?) – там почти все атрибуты можно применить к целому тексту. Почти. Но не все.

5 Самое смешное: Undo при операциях с текстом практически не работает. Они – эти операции – вообще как-будто идут у *Scribus*'а «вне конкурса» – более ранние манипуляции с самими фреймами отменить можно, а действия над содержимым этих же фреймов (если там текст) – нельзя. Я впрочем, догадываюсь, почему: *Scribus* воспринимает весь текст как массу объектов, каждый из которых – отдельный знак. Это видно, когда вы запускаете операцию замены шрифта на достаточно большом массиве текста – идет она очень долго, зато в конце вы узнаете, сколько же в вашем фрагменте знаков… Впрочем, о функции замены вспомним позже – на более приятном фоне.

6 *Scribus* априори русифицирован и умеет делать русские переносы, однако качество то ли словаря, то ли алгоритма их расстановки оставляет желать много лучшего. И все же это лучше, чем ничего.

7 Каждый акт форматирования текста требует от вас подтверждения (если вы правите во встроенном редакторе, ибо так все же удобнее) и терпения, так как *Scribus* будет долго перерисовывать экран. Секунды, но они складываются в часы пустого и раздражающего ожидания. Windows-версия и тут проявила себя в более выгодном свете, но все же комфортной такую верстку не назовешь.

После всего сказанного даже как-то нелепо добавлять, что вас ждут (и с этого надо было начинать!) ожидаемые проблемы со шрифтами. Подключив, что можно, с помощью рекомендаций Ника Вейча или «всемирного разума» из Сети, вы обнаружите, что… да-да-да – дьявол создал кодировки. *Scribus* опять-таки идеологически безупречен – он работает с полноценными Unicode-шрифтами без малейшей запинки. Проблема в том, что шрифтов с верно расположенной – по Unicode-меркам – кириллической частью в обращении не так много. Даже свежекупленные нами гарнитуры ParaType распознавались исключительно в части латиницы. Зато полностью отображались стандартные шрифты Windows – Arial, Tahoma, Impact и Times New Roman, а также (ура!) Red Hat Liberation Fonts (см. врезку *«Сеть в помощь!»*). Вы, наверняка, обратили внимание, что наш подопытный материал отличается от остального журнала. Так вот – не от хорошей жизни.

Но вернемся в *Scribus*. Большая часть манипуляций объектами в нем ведется при помощи панели Свойств. Надо ли говорить, что это не самый удобный способ для верстальщика? Возможности визуального макетирования страницы вольно или невольно сведены к минимуму – вставленную картинку куда проще отмасштабировать цифрами в окошках, нежели при помощи обычной мыши. В принципе, отсюда один шаг до верстки в текстовом редакторе при помощи тегов разметки (это и InDesign, в общем-то может), но во-первых, для этого есть LaTeX, а во-вторых, «нормальный» верстальщик мечтает о другом.

Таскать объекты тоже крайне неудобно, ибо они регулярно упираются в невидимую «стенку» и отказываются «тащиться» дальше. Приходится по несколько раз «перехватывать» их мышью (отпустил – снова взял, уже за другое место). Редактирование контуров и узлов (а также обтекания объектов текстом) – вполне функциональное, но опять же весьма неудобное и неинтуитивное. Впрочем, здесь и не каждый векторный редактор окажется на высоте – скажем прямо. Так что по этой части нас, будем считать, все устраивает.

В *Scribus* крайне неудобно реализована работа с направляющими разметки колонок и шаблонными (master) страницами: описать коротко это сложно, но раздражает сильно. Особенно, когда *Scribus*, допускающий возможность начать файл как с правой, так и с левой страницы разворота, начинает (в последнем случае) «про себя» путать их местами, рассчитывая колонки для левой полосы, исходя из полей правой и наоборот (проблема актуальна, если внутренние и внешние поля полосы на развороте различаются по ширине). К тому же колонки разделяются парой обычных, независимых – а не сцепленных – направляющих. Мелочь, а тоже требует лишнего внимания и аккуратности.

При всем этом обилии неудобств и странностей, повторюсь, *Scribus* можно считать во-первых, состоявшимся, а во-вторых – вполне неплохим продуктом. Опять парадокс? Отнюдь. Конкурентов у него нет, но он и сам с усами: мы все же сверстали в нем эту чертову статью! И вы сверстаете в нем все, что сочтете нужным, как сделали это мы.

Разработчики, похоже, не просто так «недодбели» по части работы с текстом. Тут и идеология, о чем я уже упоминал: одна задача – один инструмент. Тут и другой феномен *Scribus*: судя по всему, его авторы решили сразу отстроиться от других систем верстки, сосредоточившись на их слабых местах. Именно там *Scribus* и сосредоточил свои очевидные плюсы: мощный диалог поиска и замены текста, объектов и атрибутов; быстрый префлайт; модуль коротких слов, обеспечивающий неотрывание предлогов и союзов, а также единиц измерения и сокращений от соседних слов (это одно из требований профессиональной верстки); здесь удобно менять уровень объекта в стопке на монтажном столе (не путать со Слоями, которые тоже есть), работать с помещенными в верстку изображениями и ориентироваться во всем множестве монтажных единиц верстки благодаря специальному менеджеру объектов. *Scribus* скромен в потреблении оперативной памяти вашего ПК и не склонен отвлекать вас никчемными вопросами, не забывая, однако, предупредить о возможных проблемах. Кстати, вы найдете в нем работающими многие знакомые сочетания горячих клавиш. И все это радует.

Очень длинно получилось? Хотите то же самое – коротко? Прошу!

Ваше мнение

Вы знаете, как решить проблемы, ставшие для нас камнем преткновения? Или, может быть, располагаете солидным арсеналом приемов работы в Scribus, которые бы сделали жизнь Родиона куда проще? Присоединяйтесь к обсуждению этой статьи на нашем форуме: www.linuxforum.ru

LINUX FORMAT Рейтинг 5/10

Scribus 1.3.3.9 www.scribus.net **Лицензия: GPL**

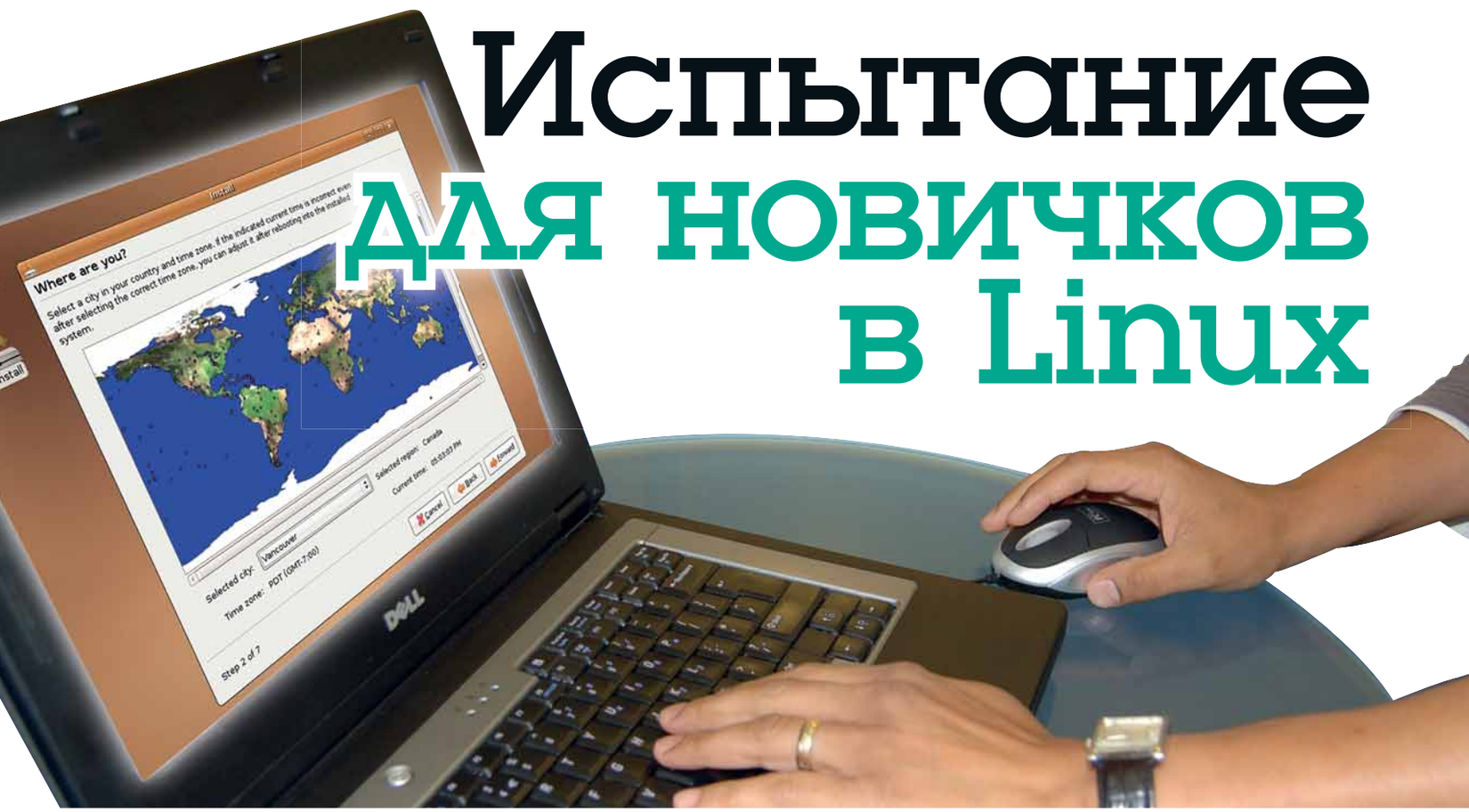
Возможности	8/10	Совместимость	5/10
Функционирование	5/10	Документация официальная	4/10
Удобство использования	4/10	Документация неофициальная	9/10

» *Пока Scribus хорош только для простой верстки – вы запросто сделаете в нем листовку или книгу, может быть отчет или даже презентацию. А вот для проектов с богатым форматированием, сложным макетом, таблицами и высокими требованиями к шрифтовому дизайну, каковыми сейчас является большая часть периодических изданий он еще не созрел… Вы, конечно, сделаете в нем и это. Но старый добрый Aldus PageMaker 5.0 образца 1994 года, не сильно уступив Scribus в функциональности, а кое где и превзойдя его, сделает ту же работу раз в десять быстрее.*

Зато теперь мы можем с уверенностью сказать, что день, когда LXF полностью перейдет на открытый издательский процесс, должен будет войти в историю как долгожданный День победы Linux на рабочем столе! LXF

» Любите читать о мучениях новичков Linux в вашей любимой ОС? Переходите к стр. 42!

Испытание для новичков в Linux



«Этот год, безусловно, станет годом Linux на рабочем столе». Мы слышим это снова и снова. Но готов ли Linux для массового использования? Это и выясняет **Майк Сондерс**.

Когда-то мы все были новичками. Мы помним свои первые вторжения в мир Linux – кропотливое изучение рабочего стола, исследование файловой системы и первые шаги в командной строке. Несколько лет назад казалось, что Linux находится на заре новой эры и готов бросить вызов рабочим столам Microsoft и Apple с их простой установкой и удобным рабочим окружением. Но достигли ли мы этой цели на самом деле?

Анекдотические свидетельства повествуют о бабушках, чувствующих себя в Gnome как рыба в воде, детях, осваивающих *Synaptic* и пожизненных пользователях Windows, которые внезапно понимают, как работает Linux.

В то же время мы видим и сбитых с толку новичков Linux, которые считают, что эта операционная система слишком сложна для повседневного использования. Нам нужно понять все их «как», «почему» и «что» – как они используют Linux, почему считают его сложным и что можно сделать, чтобы им помочь. Итак, здесь в наших Башнях LXF мы решили поставить опыт. Мы выбрали трех пользователей с различными навыками, которые никогда не использовали Linux всерьез, и решили посмотреть, как они возьмутся за эту ОС в 2007 году.

Мы посадили каждого из них перед компьютером со свежим дистрибутивом Ubuntu и присоединили к их голове электроды (шутка). Мы просили их выполнить различные повседневные задачи и смотрели, как они это делают и с какими трудностями сталкиваются, будучи пользователями других ОС. Наши задания были распределены на 3 группы:

» **Базовое использование.** Обычные действия, которые пользователям Linux приходится выполнять каждый день. Сюда входит запуск браузера и текстового процессора, копирование и удаление файла и запись CD-ROM.

» **Администрирование.** Действия, которые требуют немного больших усилий и затрагивают настройки системы: установка новой программы, изменение времени и настроек клавиатуры и выполнение обновления системы для получения последних пакетов и исправлений по безопасности.

» **Установка.** Это крупная задача, ужасающая новичков, которые не могут обратиться за помощью к опытному пользователю Linux. Подразумевается инсталляция Ubuntu 7.04 в обычной конфигурации без двойной загрузки, чтобы избежать лишних трудностей.

Кого же мы выбрали для прохождения этих тестов? Термин «новичок» здесь очень широк и охватывает как полных новичков, так и опытных пользователей Mac и любителей поковыряться в настройках Windows, которые знают реестр как свои пять пальцев.

Тем не менее, все они новички в Linux. Итак, мы выбрали по одному человеку в каждой из этих категорий и подвергли их нашим испытаниям, отмечая их успехи и обращая внимание на возникающие камни преткновения.

Подобное исследование позволило проникнуть в суть образа мышления новичков Linux и выявило конкретные изменения в вопросах удобства использования, которые необходимо произвести. Готов ли Linux для Джо Дилетанта? Означает ли «монтирование» что-нибудь, если речь идет не о кондиционерах и пластиковых окнах? И может ли наш художественный редактор полчаса не ругаться? Смачные подробности чуть ниже...

Новичок

Ее вотчина – *Outlook* и *Word*.

› Хелле Петерсен работает помощником издателя и, по ее собственным словам, она — полный новичок в компьютерах. На работе ей каждый день приходится пользоваться электронной почтой и офисными программами, но вне ее она лучше сразится с акулой-убийцей, чем проведет три секунды в командной строке.

Для Хелле компьютер – всего лишь средство, которое помогает сделать работу быстрее, а не инструмент для хакерских штучек. Она представляет миллионы пользователей, которые всегда выполняют одни и те же задачи. Для них компьютер – помощник, который не требует глубокого изучения. Хелле редко использует что-то кроме обычных приложений: web-браузера, клиента электронной почты и текстового процессора.

Но у тех, кто работал только с Windows, есть определенные ожидания. Когда мы попросили Хелле запустить web-браузер, она сразу кликнула на иконку *Firefox* на верхней панели. Но запустить текстовый процессор оказалось сложнее. «Где *Word*?», – спрашивала она и долго бродила по меню, перед тем как запустить *Evolution*. Выбор Microsoft такого общего названия для своего продукта заставляет многих думать, что текстовые редакторы на любой платформе называются именно так. Однако после некоторых поисков она запустила *OOo Writer*.

Работа с файлами породила большие проблемы. Когда мы попросили Хелле скопировать существующий файл на рабочий стол, она открыла контекстное меню и, просмотрев все пункты, выбрала Копировать (Copy). Здесь у нас есть команда, который ничего не делает: надеясь отдать компьютеру приказ, она выбрала Копировать, и что произошло? Совершенно ничего. «Я просто нажала Копировать, но ничего не произошло», – жаловалась она.

Как мы знаем, команда Копировать означает «скопировать файл в буфер обмена для последующей вставки», однако для Хелле она означала незамедлительное копирование файла. Этот недочет заставил Хелле залезть в меню Отправить (Send To) и Приложения еще до того, как я объяснил ей механизм копирования и вставки. Для новичков пункт контекстного меню, не порождающий никакого ответа системы, крайне непонятен. Не легче нам далось и удаление файла. Не найдя пункт Удалить в контекстном меню («Удаление всегда по правой кнопке!»), она открыла файл в текстовом редакторе *gEdit* и попыталась найти этот пункт в меню Файл (File). Однако, когда я объяснил ей, как работает Корзина, она очень быстро сориентировалась и спокойно удалила файл. Так что эти проблемы можно отнести к плохо спроектированному контекстному меню.

> **«Почему бы не сделать это более понятным для людей, которые не знают жаргона?»**

Головоломка с клавиатурой

Нашей следующей задачей было изменение раскладки клавиатуры. Сначала Хелле зашла в меню Приложения, а минуту спустя – в меню Система и стала изучать его содержимое. Она была озадачена огромным количеством пунктов меню размером почти в высоту экрана в 1024x768, но в конце концов нашла настройки клавиатуры («Я даже их не видела!»). Хелле никогда не записывала диски раньше, поэтому мы решили пропустить этот шаг и перейти к установке Ubuntu. Первые несколько окон Хелле прошла неплохо. Во время выбора места жительства датчанка Хелле попыталась найти свою родину в списке и, сделав это, получила длинный перечень городов, из которых нужно было выбрать свой. Она не знала, что можно указать город, щелкнув на карте мира и увеличив ее масштаб. Это неудивительно, так как установщик Ubuntu совершенно ничего об этом не говорит! Простые всплывающие подсказки были бы полезным дополнением и понравились бы всем независимо от уровня владения языком.

Следующей преградой было разбиение диска на разделы. Хотя концепция разбиения диска не была для нее совершенно чуждой, она все же спросила: «Зачем нам это делать?» и уперлась в меню Разбиение (Partitioning), как в стену. Установщик Ubuntu не предоставляет ни капли информации – он сразу предлагает выбрать способ разбиения диска (автоматически или вручную), и об остальном приходится догадываться. Хелле выбрала автоматическое разбиение, хотя легко могла и остановиться, увидев сложное окно разбиения вручную и ужаснувшись его содержимому. Несомненно, установщику Ubuntu нужен хотя бы абзац текста, поясняющего, что такое разделы и какой вариант выбрать начинающим пользователям.

Во всем остальном установка продолжалась безо всяких проблем. Сначала Хелле была несколько ошеломлена перспективой работы в чуждой операционной системе, но несколько подсказок помогли ей понять основы. Хотя Хелле с ее опытом предстоит еще несколько важных уроков (особое внимание нужно уделить контекстным меню *Nautilus* и разбиению диска на разделы), она дала нам надежду, так как ни разу за все время не растерялась. Хорошее начало...

Человек Mac

Может ли он начать думать по-другому, чтобы использовать Linux?

Мы говорим «Mac OS X» и подразумеваем «дружественный пользовательский интерфейс». Многие разработчики Gnome восхищаются оформлением этой ОС, несмотря на то, что Apple иногда не соблюдает собственных рекомендаций по проектированию интерфейса! Эффи, художественный редактор **LXF**, работает в Mac постоянно, и хотя мы не можем отнести его к опытным пользователям (командная строка для него – terra incognita), он уверенно работает с окнами и файлами. Его знания о Linux ограничиваются обработкой сотен снимков экрана, которыми мы заваливаем его каждый месяц. Вот так.

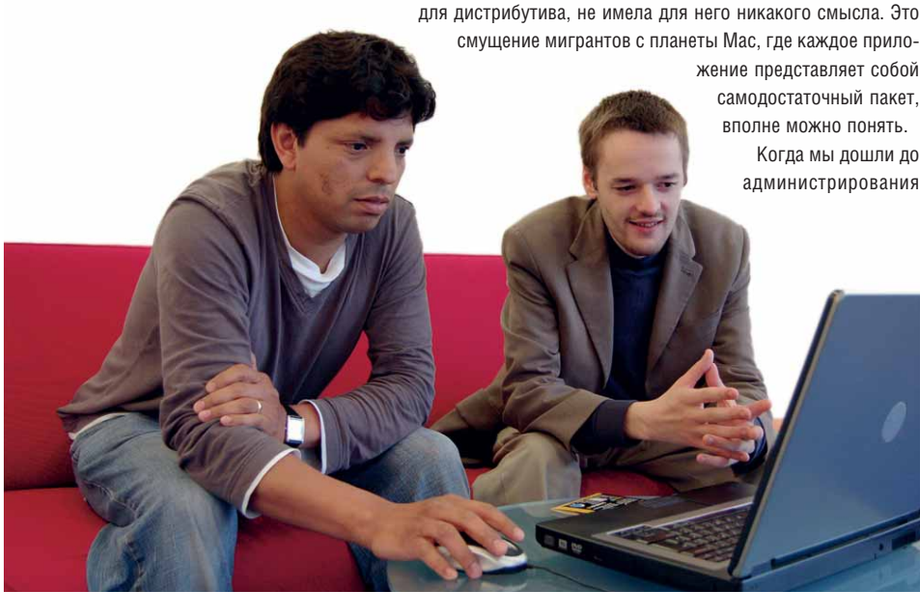
Начало было многообещающим: Эффи сразу нашел *Firefox* и *OOo* – программы, которыми он раньше пользовался в OS X. Когда мы попросили его скопировать файл, его несколько смутил пустой рабочий стол Ubuntu. Эффи не мог понять, куда сбросить копию. Он поискал диск или каталог, но когда это не помогло, решил сменить тактику и покопаться в контекстном меню. Копирование и Вставка сделали свое дело.

Похожие трудности возникли при удалении файлов. Эффи потратил почти минуту на поиски некоего устройства удаления файлов, затем я спросил его, искал ли он корзину. «Да, но здесь ее нет!» – последовал ответ. Апплет корзины в Gnome просто слишком маленький, чтобы его заметить, и к тому же спрятан в самом углу экрана – полная противоположность большой серебряной иконке в OS X. В продолжение этого шага я попросил Эффи изменить размер панели, содержащей апплет корзины. Он сразу кликнул правой кнопкой мыши на панели, выбрал Свойства и увеличил ее высоту до более приемлемой. Таким образом, хотя настройки панели по умолчанию были далеко не идеальными, изменить их не составило труда.

Проблема установки

Следующим шагом была установка программы. Эффи нужно было установить *Thunderbird*, который он нашел с помощью пункта Установка/Удаление программ (Add/Remove) меню Приложения (Applications). Войдя во вкус, он заметил, что нужно поставить галочку рядом с названием программы, а не просто выделить его, как потом сделает наш опытный пользователь Windows. Однако идея пакетов и репозиториев прошла мимо него; Эффи воспринял диалог Установка/Удаление программ как ссылки на загрузку программы с сайта. «Это ссылка на сайт *Thunderbird*?» – спрашивал он. Идея репозитория, специфичного для дистрибутива, не имела для него никакого смысла. Это смущение мигрантов с планеты Mac, где каждое приложение представляет собой самодостаточный пакет, вполне можно понять. Когда мы дошли до администрирования системы, для изменения раскладки клавиатуры Эффи сразу зашел в меню Система и вскоре запутался в утилите настройки клавиатуры. Это уму непостижимо, но в программе оказались вкладки Раскладка (Layout) и Настройки раскладки (Layout Options). Эффи выбрал последнюю и начал ломать голову. «Здесь нельзя изменить раскладку!» – недоумевал он, поэтому я направил его на другую вкладку, где он и нашел список – очень маленький (в экранном смысле) список раскладок, отнюдь не кричащий: «Меня можно сделать больше!». Мы сообщили об этих недостатках команде разработчиков Gnome. Когда мы записывали CD, Эффи поставило в тупик то, что нужно было открыть меню Places. «Ничего не понятно», – огорчался он. Однако, как только он нашел его, дело пошло намного быстрее. Установка Ubuntu тоже прошла безупречно, замедлившись только однажды: так как Эффи ни разу в жизни не видел двойной загрузки, он не увидел потребности вручную переразбивать «том» (термин из Mac OS). Как только установка была закончена, Эффи наградил тест добрыми словами, вспомнив, как долго ему приходилось рыться на рабочем столе, чтобы что-нибудь найти. Впрочем, он отметил, что со временем к этому можно привыкнуть. «Графически, Linux может быть очень простой ОС», – добавил он – лучший комплимент от пользователя Mac, – хотя тут же уточнил, что «для новичка временами он все же недостаточно нагляден». Все закончилось парой неуместных ругательств.

Эффрейн Эрнандес-Мендоса – художественный редактор *Linux Format* и фанат Astro Boy. Когда Эффи не ругает футбольную сборную Мексики, он макетирует страницы LXF в Mac OS X. Он на короткой ноге с OS X и ее приложениями. Он никогда не углубляется в сложные компьютерные темы, но каждый день использует Mac и имеет базовые навыки работы в Windows.



Итоги

Все прошло хорошо. Мы были приятно удивлены тем, что ни у одного из участников не было заметных проблем с установкой и им потребовалось всего несколько попыток, чтобы получить навыки работы в Gnome. Несомненно, есть недостатки, которые нужно исправить – контекстные меню рабочего стола и недостающий текст в программе установки соответственно – однако во всем остальном Linux не создал им заметных препятствий.

Сегодня пользователи Mac OS и Windows вполне могут пробовать Linux. Конечно, начав использовать его на постоянной основе, они

Опытный пользователь

Он запускает Regedit с закрытыми глазами…

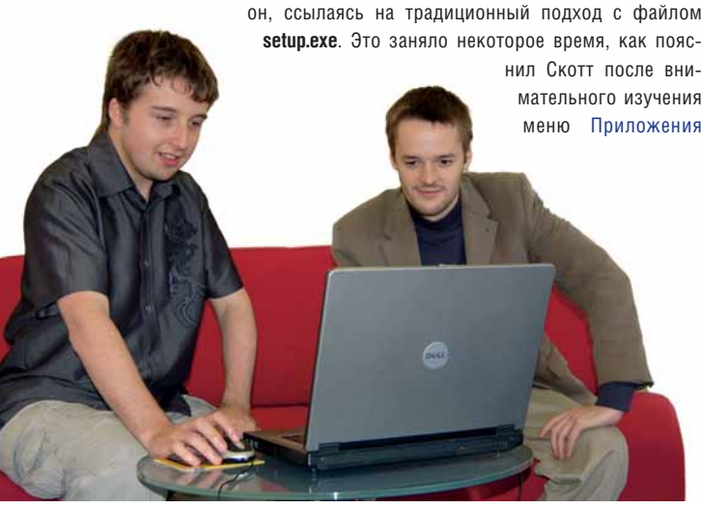
Скотт Юэрт – постоянный пользователь Windows, который работает в журнале PC Answers. Он – свой парень в Виндоусвилле и может объяснить, как заменить текст кнопки «Пуск» на альтернативный. Художник, web-мастер и иногда любитель попрограммировать, Скотт представляет типичного опытного пользователя Windows, который знает о растущей популярности Linux.

Нашим последним подопытным кроликом был Скотт Юэрт, вежливый редактор диска журнала **PC Answers**, побратима **LXF**. Как и многие постоянные пользователи Windows, Скотт все чаще слышит о Linux, но еще не пробовал его в действии, если не считать просмотра экранных снимков и периодических игр с уже установленной копией на чужом компьютере. Он из тех пользователей Windows, которых многие линуксоиды стремятся склонить на свою сторону: хорошо знаком с техническими деталями, хочет осваивать новое, но часто повторяемые утверждения о том, что Linux еще не готов для использования на рабочем столе, мешают ему начать.

Первым, что мы заметили, была склонность Скотта к использованию горячих клавиш; в то время как остальные два участника исследовали рабочий стол Linux только с помощью мыши, Скотт решал наши задачи с помощью совершенного, интуитивного владения клавиатурными комбинациями. Когда мы попросили его скопировать файл, он выделил его и быстро нажал Ctrl+C и Ctrl+V. Чтобы удалить файл, он выбрал его и нажал Shift+Delete, известное сочетание клавиш Windows, которое полностью удаляет файл (вместо перемещения его в корзину). «Shift-delete – необратимое удаление файла», - уверенно сказал нам Скотт. Однако этот образ мышления пользователя Windows проявил себя и когда дело дошло до установки приложения: Скотт не знал о существовании в Linux менеджера пакетов и полез на сайт *Thunderbird*, когда мы попросили его установить эту программу.

Трудности с пакетами

Все вполне логично – вы можете загрузить сборки *Thunderbird* для Linux с сайта проекта. Но чтобы объяснить, что такое менеджер пакетов, мы должны были направить Скотта в верном направлении. «Это немного непривычно – в Windows вы обычно находите файл, из которого и устанавливаете приложение», – сказал он, ссылаясь на традиционный подход с файлом **setup.exe**. Это заняло некоторое время, как пояснил Скотт после внимательного изучения меню Приложения (Applications): «В Windows, Установка и удаление программ чаще используется для удаления программ, чем для их установки».



Здесь нас ждала другая проблема: после того, как Скотт нашел приложение, выделил *Thunderbird* в результатах поиска и нажал кнопку Применить, его ждала полная тишина. Средство установки и удаления программ в Ubuntu требует, чтобы вы отметили галочкой каждый пакет, который хотите установить. Это показалось Скотту непривычным, учитывая то, что он щелкнул и выделил необходимую программу. Несомненно, это недочет интерфейса, который нужно исправить. «Ничего не произошло!» – отчаянно воскликнул Скотт. Для записи диска Скотт предполагал использовать отдельную программу и был несколько смущен тем, что ссылка CD/DVD Creator была спрятана в меню Places. Впрочем, когда он нашел ее, то без труда записал диск, переместив необходимые файлы мышью прямо в окно браузера. Администрирование системы тоже не вызвало особых трудностей. Скотт изменил раскладку клавиатуры и системное время, причем он знал, что щелчок правой кнопкой на последнем вызывает диалог настройки. Это другой пример того, как приемы работы в Windows оказываются полезными в Linux.

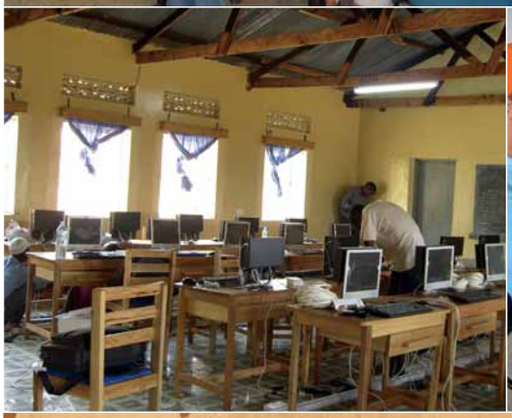
Последним шагом стала установка Ubuntu. Скотт знал, что такое разбиение диска на разделы, так что это не стало ему помехой. Однако он столкнулся с той же проблемой, что и два предыдущих участника: при выборе места жительства он не понял, что можно щелкнуть на карте и затем увеличить ее масштаб. Вместо этого он несколько минут листал список, проклиная низкую скорость прокрутки.

Программе установки Ubuntu в самом деле нужно как-то выделить возможность изменять масштаб карты! Дальнейшая установка не вызвала у Скотта совершенно никаких трудностей. LXF

встретятся с трудностями. Установке программ мешают ограничения менеджеров пакетов: вы можете установить необходимую версию программы из репозиториев своего дистрибутива, но если вы захотите обновить ее, то придется либо воспользоваться пакетами с сайта проекта (если они существуют), либо ждать обновления целого дистрибутива. Это не относится к Gentoo и другим подобным системам, но их сложно освоить новичкам.

В то же время опытные пользователи Windows, такие как Скотт, все еще должны бороться с проблемами совместимости последнего крутого гаджета. Эти проблемы нужно решать, но после этих тестов мы знаем, что Linux вполне достаточно для решения большинства задач новичков – управления файлами, серфинга и работы с документами. Сегмент рынка начинающих пользователей вполне готов к Linux, следующим шагом станет удовлетворение потребностей наиболее активных пользователей Windows.

Расскажите нам о своем опыте новичка или пользователя, который уже выбрал Linux: напишите несколько слов по почте **letters@linuxformat.ru** или поучаствуйте в дискуссии на форумах **www.linuxforum.ru**.



Свободу — народам

Общественные организации всего мира пользуются свободным ПО для объединения усилий, обретения независимости и наработки опыта в сфере IT. Ребекка Смолли изучает положение дел.

Лагеря для перемещенных лиц в Гулу, на севере Уганды, находятся в прифронтовой полосе. Около двух миллионов людей живут здесь в бедности и опасной изоляции. Но начиная с апреля, семь таких лагерей обрели жизненно важную связь с внешним миром. Затратив примерно $150 000, американская некоммерческая группа Inveneo, с помощью базирующейся на открытом ПО сети с питанием от солнечных батарей, предоставила более чем 100 000 человек доступ к VoIP и Интернет. Теперь больницы и школы могут связываться с соседями, донорскими организациями и медиа-ресурсами.

Развернутый на самом стыке общественных проблем и информационных технологий, этот проект является ярким примером использования свободного и открытого ПО (FOSS) в развивающихся странах. Другим примером может служить государственная политика по FOSS в Бразилии, Китае и Индии. Эти страны гораздо богаче Уганды, по оценке Всемирного банка и ООН они относятся к странам со «средним доходом, умеренно развитым». Но все четыре страны в той или иной мере исключены из общемирового процесса разработки открытого ПО. И все же, эти и другие страны пользуются открытым ПО для доступа к информации, развития информационных технологий и утверждения своего суверенитета.

Народ и государство

Далекие от угандийских лагерей трущобы Сан-Паулу, в Бразилии, также пожинают плоды компьютерных технологий на основе FOSS. В 2001 г., осуществляя свою стратегию развития города, социолог Серхио Амадеу [Sérgio Amadeu] приступил к оснащению районов «фавел» [favelas – трущобы] центрами связи (telecentro), где любой желающий мог получить доступ к Интернету и базовые компьютерные знания. Местные жители занялись делом, уровень преступности упал, начали развиваться предприятия. «Такой вид компьютерного просвещения имеет высокую социальную значимость», утверждает Карлос Моримото [Carlos Morimoto], автор бразильского дистрибутива Kurumin. «Многие из тех, кто работает сейчас программистами

и компьютерными техниками, начинали когда-то в одном из компьютерных залов telecentro, в школе или в общественном центре». Позднее бразильское правительство предложило Амадеу работу по реализации своей новой политики в отношении FOSS.

Исабела Диас Фернандес [Isabela Dias Fernandes] тоже участвовала в инициативах бразильского правительства, связанных с открытым ПО, включая проект Digital Inclusion. Она рассказывает: «Правительство предоставляет отдаленным регионам спутниковый интернет-доступ и компьютер – как сервер сети LTSP [Linux Terminal Server Project]. А наша работа – добраться до места и научить людей всем этим пользоваться».

Политика по FOSS применяется с 2003 г., и переход правительственных учреждений на Linux и оснащение школ Linux-компьютерами обеспечили бразильскую «цифровую революцию (*revolutión digital*) сверху». Но многие бразильские хакеры утверждают, что влияние подлинно народного движения было ничуть не меньшим. «По-моему, государственная политика – это скорее популистская акция, чем реальная поддержка Linux в Бразилии», говорит Моримото. «Linux процветает благодаря усилиям многих компьютерных специалистов, пользовательских групп и отдельных пользователей, стимулирующих использование Linux, создающих пилотные и переходные проекты и предоставляющих бесплатную поддержку».

Патриархи бразильского программирования стали почти культовыми фигурами. Элио Кастро, разработчик KDE и Mandriva, рассказывает: «Мы помним, как Серхио Амадеу, Марсело Бранко [Marcelo Branco] и другие прогрессивно мыслящие люди собрались в Рио-Гранде-до-Сул, чтобы обсудить политические и технические вопросы развития открытого ПО. Не все разделяли их идеи, но именно тогда было положено начало FISL (бразильскому OSCon), ежегодно собирающему более 4 000 человек».

Во всем мире наблюдается общественное, неформальное распространение открытого ПО. Исследования, проведенные в Юго-Восточной Азии профессором Джошуа Минделом [Joshua Mindel] и его коллегами из Государственного университета Сан-Франциско, показали отсутствие корреляции между государственной политикой в отношении FOSS и скоростью его распространения. В индийском штате Керала Linux установлен на компьютерах в 2 600 школах, в Тамил-Наду в сельские школы пришли 12 000 Linux-машин. Но Дж. Нагарджуна [G. Nagarjuna], председатель Индийского фонда свободного ПО (Indian Free Software Foundation), считает, что это результат усилий поборников открытого ПО, которые вели переговоры как с программистами, так и с представителями различных организаций, государственными чиновниками и учителями.

Это хорошая новость, так как подлинно народные движения свободны от лоббирования программных корпораций, которому подвергаются правительства, и не подвержены смене курса после очередных парламентских выборов.

Мост через «цифровую пропасть»

Вместе с тем, региональные и федеральные правительства стран третьего мира отчаянно стремятся расширить доступ к информационным ресурсам и преодолеть «цифровую пропасть», отделяющую их от Запада, ведь давно уже очевидно, что это ключ к процветанию наций. Еще в 1978 ЮНЕСКО учредила орган, предназначенный для выравнивания уровней богатых и отсталых стран в области технологий. Тогда же International Telecommunication Union, орган ООН, принял заявление, в котором утверждалось: «общедоступный и надежный доступ к телекоммуникационным сетям является неотъемлемым элементом конкурентоспособности каждой нации на мировой арене».

Профессор Миндел тоже отмечает важность доступа к компьютерной связи для экономического процветания: «Существовать без связи с остальным миром – все равно что жить рядом с автострадой, не имея на нее въезда. Как только въезд построен, появляется возможность обмениваться товарами и услугами в любом направлении со всем миром. И это сравнение справедливо для любого сектора экономики».

Большое внимание уделяется распространению компьютеров на базе Linux в школах. Это происходит в рамках благородного проекта «Компьютер – каждому ребенку» (One Laptop Per Child, OLPC). Один из мотивов этой инициативы – понимание того, что обучение детей информационным технологиям с раннего возраста является ключом к повышению общего уровня компьютерной грамотности в долгосрочной перспективе. Но анекдоты о фермерах, использующих мобильные телефоны для того, чтобы узнать цены на товары или ознакомиться с последними достижениями агрономической науки, все еще актуальны. С помощью Wire.less.dk Томас Краг [Tomas Krag], осуществив несколько проектов по настройке беспроводных сетей на основе открытого ПО, на практике изучил выгоды от надежной связи: «В странах, где немалая часть трудоспособного населения поневоле становится трудовыми мигрантами, отсутствие связи может вызывать тяжелые последствия как для отдельных семей, так и для общества в целом. Связь с домом, напротив, может вовлечь часть гастарбайтеров в содействие покинутой родине, до некоторой степени смягчив “утечку мозгов”».

› Ученики средней школы из Кангаи, Уганда, хозяйничают в сети на основе открытого ПО, предоставленной Invenео.

С нарастающей энергией, правительства и сообщества разработчиков предпринимают усилия для сокращения «цифрового разрыва». Это можно сделать и через проприетарное ПО – многие проекты так и поступают – но при пользовании открытыми программами нет необходимости заново изобретать велосипед. Кроме того, свободное ПО требует меньших затрат на приобретение, гибко и легко масштабируется.

В Испании некоторые дальновидные муниципалитеты уже давно пользуются открытым ПО для доступа к компьютерным коммуникациям. Эстремадура – сельскохозяйственный, сравнительно бедный регион (*junta*) с населением порядка 1,1 миллиона. Менее чем за десять лет здесь были переведены на Linux общественная школьная система, рабочие столы муниципалитетов и многое другое (*см. стр. 50*). Столь выдающиеся достижения стали возможными благодаря амбициозным, четко поставленным целям и наличию в администрации горячих приверженцев FOSS. «*Junta* всегда понимала преимущества и пользу »

Пиратство

Вероятно, самое серьезное препятствие на пути широкого распространения открытого ПО в развивающихся странах – это пиратство. Высокая цена закрытого ПО имеет значение для крупных компаний, но на улице взломанное настольное ПО можно купить за ту же цену, что и Linux. Для многих людей приобретение легальной версии просто исключено.

Пока в некоторых странах процветает пиратство, их рынки обречены на иностранное закрытое ПО, которое местные пользователи и организации изучают по нелегальным версиям. Как сказал Джефф Рейкс из Microsoft: «Все равно взломают – так пусть лучше взламывают наше, чем чье-нибудь еще». Хотя до полной победы еще далеко, но всемирная ассоциация производителей ПО (Business Software Alliance) уже сейчас отмечает снижение уровня пиратства в Китае и России (основных странах-нарушителях). Вместе с тем, растет международное давление на проблемные страны с целью полной ликвидации пиратства.

» открытого ПО, а все муниципальные службы пользуются свободным ПО еще с девяностых», говорит Франсиско Уэрта [Francisco Huerta], директор местного Центра инноваций. «Но поворотный момент для открытого ПО в Эстремадуре наступил в 2000 году, с принятием плана по выделению одного компьютера на каждых двух учеников средней школы. Использовать закрытое ПО в этом проекте было бы немыслимо как экономически, так и технически», подчеркивает Уэрта. «И тогда мы решили воспользоваться свободным ПО для оснащения 60 000 компьютеров в образовательной системе».

Локализация и распространение

Эти истории о помощи в несчастье и решении образовательных вопросов показательны, но они не отражают общей картины. С развитием дебатов о «цифровой пропасти» вопрос доступа неизбежно переходит в вопрос об участии. С точки зрения FOSS это означает, что сообществам в развивающихся странах хотелось бы разрабатывать собственное ПО.

Дуэйн Бейли [Dwayne Bailey] возглавляет южноафриканскую организацию **Translate.org.za**, переводческий открытый проект. Он считает, что для некоторых людей перевод программ жизненно необходим. «Мы думаем о 80% населения, не говорящих по-английски», говорит Дуэйн. «Городская статистика не в счет, ведь в бизнес-секторе английский доминирует. Реальность же такова, что большинству южноафриканцев современные технологии недоступны».

Понятно, что перевод проприетарного ПО требует переговоров с производителем, поэтому сообщества на местах, в Руанде и Южной Африке, обратились к сообществу *OpenOffice.org*. Как и другие FOSS-проекты (Ubuntu, Fedora или Gnome), OOo переведен на множество языков.

По мнению Бейли, перевод – проблема не только понимания. С ним тесно связано и культурное самосознание. «Я еще не встречал компьютерного пользователя – носителя языка африкаанс [язык германской группы, на котором говорят буры], не владеющего английским. Но знаю многих, которые пользуются нашим ПО на африкаанс. Почему? Да потому, что это их родной язык. В связи с этим возникает множество культурных мотивов для перевода. Вообразите, что вы набрали в текстовом процессоре слово Xhosa [язык кхоса, относится к группе банту]. Красные линии ошибок, змеящиеся по экрану, тут же дадут вам понять: ваш язык – отстой! Цифровые технологии – это передовая область познания, и если какой-либо язык в этой области не представлен, люди сочтут его убогим».

› Переводы ПО, полезные и для практических целей, также помогают людям обрести свою культурную самобытность.

Многие пользователи не любят, когда их принуждают пользоваться англоязычным ПО, будь оно трижды свободно. Поэтому они пользуются возможностями открытого ПО не только для перевода, но и для введения местной символики. Например, значок Интернет в Linex, локальном дистрибутиве Эстремадуры, срисован с местной птицы, а GIMP переименован в 'Zurbarán', в честь знаменитого земляка-художника [имеется в виду Франсиско де Сурбаран]. Это уже вопрос гордости за свою малую родину и выживания родного языка. В Индии умопомрачительное количество локализационных проектов работает над поддержкой массы мелких языков, охватывающих 90% населения. В Непале целая организация борется против засилья англоязычного (пиратского) ПО, выпуская собственный непальский дистрибутив, NepaLinux.

Локально или глобально?

Конечно, подобная демократия полностью соответствует философии свободы открытого ПО, но не всегда и не все проходит гладко. В прошлом году сообщалось, что прокурор обвинил мэра турецкой провинции Сур в «нецелевом расходовании бюджетных средств» после того, как тот сделал заказ на перевод Ubuntu на курдский и публично заговорил о равноправии языков.

У Сандживы Вираварана [Sanjiva Weerawarana], *Apache*-программиста из Шри Ланки, сомнения другого рода. «Я слегка против сверхлокализации», говорит он, «ведь чтобы быть конкурентоспособным на мировом рынке, необходимо владеть английским. Локализация – это здорово для приобщения деревни к передовым технологиям, но плохой способ продвигать наши кадры в мире».

Вираварана хочет прогресса своей страны не меньше, чем другие программисты-энтузиасты в странах третьего мира, но намерен добиваться этого прогресса совместно с мировой индустрией ПО. Вместе с активистами-соотечественниками он добился поразительных результатов, организовав целую мини-корпорацию *Apache*-программистов.

Вираварана прекрасно понимает, что большинство программ с открытым исходным кодом производится в США и Европе. Ян Мердок [Ian Murdock] говорил о необходимости для всех стран «строить себе будущее, не прикованное к погоне за прибылью единственной американской компании». Но ведь и Red Hat, и Sun, и Novell – тоже американские компании. Развивающиеся страны могут осваивать ПО, только принимая участие в проектах свободного ПО или создавая собственные. Вираварана нашел свое решение: мобилизовал шри-ланкийских хакеров на участие в проекте, в котором сам отлично разбирался – *Apache* – и работал над ним на английском языке.

Его новое достижение – WSO2, организация, предоставляющая web-услуги интернациональным компаниям, штат которой укомплектован шри-ланкийскими программистами. «Здорово, что у нас нашлось столько людей, способных разрабатывать свободное ПО, но что дальше? Как нам вложить свои успехи в развитие нашей страны?» спрашивает Вираварана. «WSO2 в каком-то смысле доказывает, что, подобрав удачную комбинацию технологий, кадров, навыков и прочего, можно выйти и на мировую арену».

На этом уровне видны захватывающие перспективы использования разными странами открытого ПО для строительства собственной IT-индустрии. В статье 2001 г. Анка Метиу [Anca Metiu] и Брюс Когут [Bruce Kogut] из Европейской бизнес-школы INSEAD говорили о разрыве между «наличием способностей и наличием возможностей»: толковые программисты не могут пробиться на элитный уровень из-за географических, экономических и языковых барьеров. Применяя FOSS, этот разрыв можно преодолеть.

Даже в странах с развитой IT-инфраструктурой программисты часто бывают вынуждены ограничиваться предоставлением услуг и поддержки. «В настоящее время индийские программисты в основном исполь-

Идеология открытого ПО

Многое в языковом вопросе зависит от политики. В 2005 Ричард Столлмен писал: «Зависимость от проприетарного ПО – это не развитие, а электронная колонизация». Но на самом ли деле политика левых режимов в отношении FOSS мотивируется идеологией открытого ПО? Исследователь Юджин Ким пришел к выводу, что задачи рабочей партии бразильского президента Лула да Силва действительно имеют много общего с целями открытого ПО. И все же большинство наблюдателей сходится на том, что правительство Бразилии и других левых или социалистических режимов принимает решения, исходя из экономической выгоды. Если в дело вступает политика, то социализм превращается в национализм. Это заметно по энтузиазму, с которым принимаются за локализацию, и по желанию обрести независимость от иностранного ПО.

Но Элио Кастро из Mandriva советует, отбросив политику, перенимать бразильский опыт: «Меньше разговоров – просто пользуйтесь проверенными там идеями».

Бразильское правительство проводит курс на организацию доступа в интернет для отдаленных поселений.

зуются на черной работе», говорит Дж. Нагарджуна. «Если бы они могли принимать более активное участие в разработке и сопровождении свободного ПО, то могли бы стать архитекторами и авторами, а не грубой рабочей силой для поддержки чужих программных систем».

Многие программисты проникли в индустрию через добровольное участие в разработке свободного ПО: например, Диас Фернандес [Dias Fernandes] – Linux-программист-самоучка. Имея интернет-доступ, любой хакер без труда может участвовать в списках рассылки и почтовых дискуссиях, и входной барьер для новичка в свободном ПО намного ниже, чем в частном, с его формальным, дорогостоящим обучением.

Чем больше людей научится работать с Linux, тем больше участников из развивающихся стран будет в проектах свободного ПО. «Обратите внимание», говорит Дейв Нири [Dave Neary] из команды Gnome, «насколько вырос объем инвестиций различных компаний в азиатские проекты – у Novell есть команда в Бангалоре, работающая над Evolution, Sun в Китае набрала группу для разработки Gnome; список можно продолжить. Такие связи помогают идеям свободного ПО укорениться в этих странах».

Но этот процесс требует времени. Дж. Аарон Фарр [J. Aaron Farr] руководит web-проектом JadeTower («Нефритовая башня») в Китае и продвигает там открытое ПО, в частности *Apache*. «Открытые, коллективные, беспристрастные методы работы Open Source плохо приживаются в китайском бизнесе и IT-индустрии», говорит он. «Допустим, два программиста, Адам и Боб, одновременно и публично предложили заплатки по одному и тому же вопросу в открытый проект. Естественно, будет выбрано технически более грамотное и элегантное решение. В китайском же обществе, если решение Боба будет худшим, зато он будет занимать более высокое общественное положение или иметь более тесные связи с руководителями проекта, отклонить его заплатку будет социально весьма нелегкой задачей».

Опасность разделения

Многие приверженцы FOSS в развивающихся странах предпочитают глобальное объединение усилий раздробленности стран, строящих собственные проекты. Они опасаются, что свободное ПО может разделиться на много мелких ответвлений, которые утратят совместимость. В январе этого года Linux.com сообщал о нарушениях GPL в Бразилии, где некоторые проекты отказались опубликовать исходные коды и разработали собственные, менее альтруистические лицензии. По словам Карлоса Моримото, это просто недоразумение, а исходный код Kurumin Linux всегда был доступен онлайн. Представитель Linux Foundation, организации, которая следит за соблюдением стандартов, Джим Землин [Jim Zemlin], считает все это бурей в стакане воды. «Нарушения GPL? Да это ничтожнейшая из их проблем, поверьте. По мне, могли бы замахнуться и на большее! Возмещение? Кому оно нужно? Да, да, конечно, GPL нужно чтить, но… я бы взглянул на это шире».

Гораздо больше Землин озабочен соблюдением интернациональных стандартов. «Велик соблазн объявить, в духе протекционизма, о создании собственных стандартов где-нибудь в Таиланде или в Китае. Но из этого ничего не выйдет: ведь ни один разработчик приложений не захочет приспосабливать свой код к экзотической версии Linux для микроскопического рынка в третьем мире».

Необходимость поддержки

В долгосрочной перспективе для поддержания жизнеспособности проектов FOSS в развивающихся странах очень важно создание локальных групп Linux-программистов. Например, в Кении открытое ПО продвигается очень туго – отчасти из-за отсутствия локальных компаний, способных оказать поддержку, отчасти из-за продолжающегося использования проприетарного ПО в системе образования. Доркас Мутхони [Dorcas Muthoni], руководительница фирмы в Найроби, которая консультирует правительство и предприятия по вопросам открытого ПО, утверждает: «OSS откроет людям и предприятиям неограниченные возможности, но только в том случае, если кто-нибудь возьмет на себя инициативу по поддержке OSS, и люди будут знать, что их не оставят в трудную минуту».

Без поддержки переход на Linux может быть провален, и организации вернутся к закрытому ПО. Inveneo, развернувшая сеть в северной Уганде, недавно организовала курсы подготовки IT-профессионалов для Западной Африки в поддержку деревенских проектов. «Приходилось без конца летать в Уганду, чтобы поправить то там, то сям», рассказывает Лаура Меллоу [Laura Mellow], исполнительный директор Inveneo. «Наш опытный образец был огромным ящиком, и в один прекрасный день его прогрызли крысы и поселились внутри. Ясно, сказали мы, пора подумать о местных сотрудниках». Теперь, при финансировании от AMD, администраторов обучат Inveneo Linux (дистрибутив на базе Xubuntu) и другим открытым и VoIP-технологиям, обеспечат им бизнес-поддержку, и тогда они, наконец, смогут отработать затраченные вложения.

Стремление к достатку

Во всем мире различные сообщества пользуются открытым ПО для того, чтобы поправить свое финансовое положение, будь то в угандийском лагере беженцев или испанской школе. Пользователям из развитых стран, привыкшим считать открытое ПО альтруистичным по природе, упор на экономический подтекст кажется приземленным. Кристиан Эйнфельдт [Christian Einfeldt], который фиксирует прогресс открытого ПО в документальном фильме под названием *Digital Tipping Point*, думает иначе: «Наша потребность в супертехнологиях до некоторой степени деловая, но также и развлекательная», говорит Кристиан. «В развивающемся мире это далеко не так: там люди жаждут свободы, благосостояния, мощи. Они устали от притеснений, от того, что их язык и их культура воспринимаются как второсортные. Они хотят быть равными среди равных, они самоутверждаются».

Диас Фернандес вторит ему: «Я тоже считаю, что источник бразильского энтузиазма кроется как раз в этом: очень многие ждали своего шанса – получить доступ к информации, изучить технологию и извлечь из этого пользу для себя». *Compiz* подождет – работать надо! LXF

Встань рядом или узнай больше

» **www.itu.int/pub/D-IND-ICTOI-2006/en** Узнай о состоянии компьютерных коммуникаций в разных странах.

» **http://geekcorps.org** Прояви свои способности в IT-проекте развивающейся страны.

» **www.digitaltippingpoint.com** *Digital Tipping Point* – это гигантский документальный кинопроект, включающий интервью со многими активистами FOSS во всем мире. Проекту нужны расшифровщики, администраторы репозиториев, директоры и др.

» **www.sahana.lk** *Sahana* – это известный открытый проект по предупреждению и ликвидации чрезвычайных ситуаций, основанный после цунами 2004 г. в Юго-Восточной Азии.

» **https://wiki.ubuntu.com/LoCoTeams** Региональные команды пользователей и активистов Ubuntu. Нет ли какой-нибудь поблизости?

» **http://code.google.com/soc** Google Summer of Code: прекрасная возможность заявить о себе для Linux-хакера из развивающейся страны.

» **www.bosco-uganda.org** Узнай больше о жизни в лагерях беженцев Уганды после получения доступа к компьютерам.

Распространение открытого и свободного ПО во всем мире

Эстремадура, западная Испания

Хотя Испанию развивающейся страной не назовешь, но Эстремадура, расположенная вблизи границы с Португалией, всегда была отсталой территорией, с самой низкой плотностью населения и самой высокой безработицей в стране. В девяностых годах местное правительство в поисках выхода из экономического тупика обратилось к сфере IT, и с этого времени началось победное шествие открытого ПО.

Вначале был план ликвидации компьютерной безграмотности, принятый в 1999 г., с целью донести компьютерные знания до каждого жителя Эстремадуры. После построения локальной сети вскоре стало ясно, что свободное ПО может и должно стать основой всех будущих проектов. Был разработан специальный дистрибутив на основе Debian, GnuLinEx, который и поныне используется в школах, государственных и медицинских учреждениях, библиотеках и общественных центрах во всей Эстремадуре.

Продюсер Кристиан Эйнфельдт посетил Эстремадуру во время съемок своего документального фильма *Digital Tipping Point*. «Они и в самом деле перевернули свое общество, и сделали это разумно и слаженно», говорит Кристиан. Серьезные исследования проводить еще рано, но Луис Миллан Васкес де Мигель [Luis Millán Vázquez de Miguel], региональный IT-министр, утверждает, что безработица в Эстремадуре падает сейчас самыми быстрыми темпами во всей Испании, а валовой продукт растет на 3,5% в год.

Кроме того, он сказал: «Налицо зарождение технологического сектора, ранее практически не существовавшего. У нас есть проект под названием Vivernet, инкубатор предпринимателей, который за пять лет вырастил 80 технологических компаний или компаний, которые встроили технологию в свой производственный процесс. Постоянно образуются малые предприятия, и занимают ниши, образовавшиеся в результате использования свободного ПО в регионе. А еще мы видим, как гиганты индустрии, такие как IBM и INDRA, приходят в Эстремадуру со своими предприятиями. В городах Касерес и Бадахос они открыли свои филиалы, трудоустроив около 500 программистов».

По вопросам разработки свободного ПО регион тесно взаимодействует с сообществами Латинской Америки. «При наших скромных ресурсах мы стараемся хорошо усвоить все уроки и освоить преимущества открытого ПО, на пользу для своего региона и всего мира», считает де Мигель. www.linex.org

Один из общественных центров открытого ПО в Эстремадуре.

NorthXSouth, Сан-Паулу

Чтобы стать участниками движения открытого ПО, а не просто пользователями готовой продукции, необходимо развивать навыки Linux-программирования. Интернет-компания NorthXSouth вливается в растущее сообщество латиноамериканских интернет-разработчиков открытого ПО, извлекая выгоду из работы с местными программистами и одновременно обучая их.

Впервые основатели NorthXSouth соприкоснулись с сообществом программистов Южной Америки через добровольческие акции, в рамках некоммерческого проекта LineFeed предоставляя IT-помощь группам активистов и сообществам в кризисных ситуациях. В дальнейшем компания выполняла заказы бразильского правительства, курируя переходы на Linux, проекты web-разработки и установку спутниковых узлов связи в отдаленных уголках Бразилии. «Мы обнаружили, что после реализации этих проектов образовалась целая сеть программистов, дизайнеров и менеджеров проектов, которые могли работать на английском, португальском и испанском языках», говорит Райан Багерос [Ryan Bagueros] из NorthXSouth. «Поэтому, помимо осуществления миграции на свободное ПО в Латинской Америке, мы взялись за корпоративную интернет-разработку в Сан-Франциско, эффективно применяя труд сторонних программистов (аутсорсинг). Так мы и действовали весь последний год, укрепляя и свой бизнес, и сеть разработчиков Developers Network, включающую свыше 20 опытных специалистов в Латинской Америке и США».

NorthXSouth и сейчас действует по обе стороны от южного тропика, предпочитая проекты по миграции на свободное ПО в Латинской Америке. Багерос убежден, что свободное ПО идеально для роста числа профессиональных программистов в развивающихся странах, так как, изучая открытое ПО, они могут многому научиться самостоятельно, а опыта можно набраться, принимая участие в добровольческих проектах. «Чтобы стать интернет-программистом, не нужна ученая степень по информатике – опыт важнее», говорит Багерос. «Проекты свободного ПО организованы так, что человек может немедленно принять участие в цикле разработки, а опыт придет со временем». www.northxsouth.com

Местные участники спутникового интернет-проекта на wiki-семинаре Конференции по открытому ПО Латинской Америки в Ресифи, 2005 г.

Ndiyo, Фултола, Бангладеш

Все мы бывали в интернет-кафе, но представьте себе кафе, где доступ к Интернету происходит через мобильный телефон! Именно такое местный предприниматель Абу Суфиан [Abu Sufian] открыл в прошлом году в Фултола, Бангладеш, отчасти как бизнес, отчасти – как эксперимент Ассоциации GSM, доказывающий возможность использования мобильного телефона для доступа в Интернет.

Экспериментальную сеть со сверхтонкими клиентами предоставила Ndiyo, кембриджская некоммерческая организация. В состав сети входят: единственный компьютер на Ubuntu, четыре рабочих станции, Ethernet-коммутатор и мобильный телефон Motorola, обладающий функцией Edge. Сверхтонкий клиент, установленный на каждой рабочей станции, назвали Nivo (Network In, Video Out).

«Интернет-кафе приобщает к Сети тех людей, которые могли вообще никогда не получить к ней доступа», говорит Джон Ноутон [John Naughton], один из основателей Ndiyo. «Но интернет-кафе, построенные по обычной PC-технологии, дороги, ненадежны, расточают ресурсы и требуют постоянного технического обслуживания». Используя систему Ndiyo, Суфиан взял лишь небольшой кредит для открытия бизнеса.

Используя подобные сети, общественные центры вполне могут обойтись устаревшим оборудованием, и даже сэкономить на его количестве. «Нагрузка от системы Ndiyo гораздо меньше, чем от сетей на обычных ПК», сказал Ноутон. «Энергопотребление Nivo составляет 3–5 Вт, в зависимости от того, используются ли клавиатура и мышь (сравните с 60–100 Вт у обычного компьютера). У таких устройств долгая жизнь, ведь обновления ПО и аппаратуры проводятся только на сервере».

Эта концепция идет вразрез с инициативами типа OLPC и ClassMate, призванными, наоборот, нести в развивающиеся страны новейшее оборудование. «Мы и OLPC возделываем одно поле, как говорится в Библии», отмечает Ноутон, «только они делают упор на обеспечение детей мобильными устройствами, а мы пытаемся выработать новую сетевую модель, пригодную для малых предприятий в странах с низким уровнем жизни». www.ndiyo.org

Интернет-доступ через мобильный телефон в Фултола.

Translate@thon, Университет Родса, Южная Африка

Южная Африка говорит на 11 языках, но в компьютерной области доминирует английский, хотя постоянно говорит на нем всего пятая часть населения. Локализация свободного ПО играет значительную роль в популяризации информационных технологий среди не-англоязычного населения. В этой сфере очень активна некоммерческая организация Translate.org: она создает инструментальные средства для перевода и южноафриканскую клавиатуру, координирует локализацию *Firefox*, *OpenOffice.org* и других программ.

В мае при поддержке Translate.org в Университете Родса была проведена 48-часовая акция 'translate@thon', в ходе которой около 60 студентов и IT-специалистов работали над переводом открытого сервиса web-почты *IMP* с английского на кхоса. Пользуясь Xhosa-глоссарием компьютерной терминологии, добровольцы перевели около 10 000 слов из 12 800. В результате марафона была создана пробная кхоса-версия программы, готовая к тестированию перед официальным выпуском.

Руководитель Translate.org Дуэйн Бейли назвал это событие вдохновляющим. «Мне кажется, что люди стойко убеждены в баснословном богатстве программной индустрии и надеются, что она легко оплатит перевод программ на разные языки. Беда в том, что этого не происходит», сказал Дуэйн. http://translate.org.za

David McGregor

Linux-лаборатория в Университете Родса была открыта в 2005 г.

Департамент науки и технологий, Претория, Южная Африка

В качестве примера, как государство может использовать открытое ПО, неплохо взять Южную Африку. Вдохновленное успехами таких организаций, как Translate.org и Shuttleworth Foundation, правительство ЮАР теперь пожинает плоды повсеместного применения открытого ПО и использует открытые стандарты для всех новых программ, разрабатываемых как для правительства, так и самим правительством. Аслам Раффи [Aslam Raffee], менеджер департамента науки и технологий ЮАР, считает, что такая политика усиливает конкуренцию и стимулирует развитие, снижая входной барьер для южноафриканских производителей ПО. Еще один мотив: «…обеспечение всеобщего онлайн-доступа к правительственным службам без запретительных тарифов, лицензионных ограничений и подобных препятствий», говорит Аслам. «Мы уже связывались с техническим директором местного филиала Microsoft по вопросу официальной поддержки ODF».

Есть надежда, что открытое ПО привнесет большую прозрачность разнообразной информации и гласность в мировом масштабе, через расширение доступа к Интернет и открытые форматы документов, такие как ODF. В некоторых странах оно даже может снизить коррупцию и помочь становлению демократии.

Что за штука... ZFS?

Пол Хадсон представляет, может быть, последнее слово в разработке файловых систем.

ZFS... Z... А может быть, это только «последняя буква в разработке файловых систем»?
Можно было бы и так сказать. Понятие «последнее слово» – это часть громкой рекламной кампании Sun.

Значит, именно Sun затеял ZFS?
Да, начиная еще с ранних прототипов, появившихся в конце 2001 г.

Как это, 2001? Или я попал не в тот номер, провалившийся в дыру времени, либо статью следовало назвать «Что за штука *была*... ZFS?»
Нет, все правильно. Ранние прототипы, конечно, устарели, но ведь разработка файловой системы серьезного уровня – процесс по определению продолжительный. Фактически, завершенный код вошел в состав Solaris – собственной Unix-системы Sun – лишь в прошлом году.

Тогда он тоже уже несвеж!
Может быть, но как раз в последние две недели наметился прогресс в переносе этой ФС на Linux, и если вы хотите удивить друзей своими техническими познаниями, то стоит поспешить.

Для того я и покупаю LXF, верно? ОК, вы меня заинтересовали. Теперь скажите: чем же так замечательна ZFS?
Идея назвать ZFS «последним словом в разработке файловых систем» пришла потому, что эта система продумана до мелочей. В принципе, она способна хранить столько информации, что можно повредиться в уме, пытаясь уяснить себе такой объем. Возьмем, например, сегодняшнего чемпиона по объему хранения данных, диск Blu-ray, на котором можно разместить до 50 ГБ. Это немало, верно? Теперь представим компьютер, который ежесекундно перемалывает объем информации, равный одному диску Blu-ray – то есть, на жесткий диск сохраняется 50 ГБ в секунду. В течение года – считая, что диск не успеет разлететься на куски – будет накоплено чудовищное количество данных, а именно 1 577 880 000 ГБ, или около полутора миллионов терабайт. Солидно, а? Ну, а теперь представим, что мы старательно сохраняли эту информацию со времени гипотетического сотворения Вселенной, что-то около 14 миллиардов лет. Перед нами уже 17 356 680 000 000 000 000 ГБ данных, величина, уже недосягаемая для постижения.

А хватит ли 17 356 680 000 000 000 000 ГБ? У меня, знаете, весьма обширная коллекция пор... з-з, MP3.
Но и я еще не закончил! 17 356 680 000 000 000 000 ГБ – очень большое число, но нас на Земле шесть миллиардов, и каждый желает сохранить свои данные в системе ZFS, поэтому умножим-ка это число на шесть миллиардов. А заодно представим, что где-то еще во Вселенной существует шесть миллиардов обитаемых миров, в каждом из которых проживают по шесть миллиардов аборигенов, и каждый хотел бы сохранить 50 ГБ/с в одной и той же файловой системе со времени образования Вселенной. Мы получим уже 6x1038 ГБ, или 600 триллионов триллионов терабайт.

> «ZFS изменила даже базовую концепцию жестких дисков: теперь это “область памяти”».

Короче, ZFS может хранить много данных?
Да нет, не просто «много» данных. Все земляне и инопланетяне за все это время смогут занять меньше одного процента от потенциальной вместимости ZFS.

Колоссально. Ну, а помимо последнего слова по объему хранения, чем еще замечательна ZFS?
На самом деле, объем – разве что самая яркая особенность на фоне нескольких важных нововведений. Руководитель проекта назвал было эту систему Zettabyte Filesystem, но вернулся к старому доброму ZFS, прикинув, что «если назвать это Zettabyte Filesystem, придется долго объяснять, что такое zettabyte, лифт успеет доехать до верхнего этажа, и все разойдутся по офисам, думая, что у новой системы просто большая вместимость. Что тоже верно, но все же не главное». Среди прочих заметных новшеств – автоматическое деление памяти на области [storage pooling], отложенное копирование [copy-on-write] и динамическое чередование [dynamic striping].

Мрак. А нельзя ли в стиле данной рубрики «разжевать» мне все это?
«Отложенное копирование» – моя любимая функция, вот с нее и начнем. Представим себе процесс перемещения файла в 500 МБ из одной директории в другую. На Linux это происходит в мгновение ока: системе проще потихоньку изменить файловую структуру, чем побайтно копировать файл на новое место, а затем удалять оригинал. Но если файл нужно скопировать – то есть обеспечить его одновременное присутствие в двух местах – Linux осуществит побайтное копирование; этот процесс, вероятно, займет несколько минут. А если файл «копируется» в файловой системе с отложенным копированием, то в действительности создается лишь ссылка на него – в обоих местах используется один и тот же исходный файл, поэтому скорость копирования столь же высока, как и при перемещении.

А если изменить копию? Оригинал ведь испортится?
Отнюдь – потому эта функция и называется отложенным копированием. В тот же миг, когда одна из виртуальных копий попытается изменить исходный файл, произойдет его реальное копирование на новое место, и все изменения будут записаны уже в эту реальную копию. Сейчас объясню подробнее. Пусть **FileA** скопирован в директории **Foo**, **Bar** и **Baz**. Сначала этот файл находится в одном месте, и пути **Foo/FileA**, **Bar/FileA** и **Baz/FileA** указывают на один и тот же файл. Теперь, если пользователь попытается изменить **Bar/FileA**, то ZFS непременно предоставит для этого независимую версию файла, изменения в которой будут невидимы для **Foo/FileA** и **Baz/FileA**. На практике, ZFS может применить эту функцию для создания версионной системы, в которой все версии всех файлов будут сохраняться автоматически.

Выходит, она сделает резервную копию моего жесткого диска?
Нет. По крайней мере, не в обычном смысле слова. Вместо одной огромной резервной копии у вас будет множество мелких, разбросанных по всей системе. Но если нужна традиционная резервная копия, то ZFS и тут поможет, ведь она

умеет создавать отложенные клоны данных. При этом вся файловая система действует как одна ссылка на отложенное копирование, то есть можно модифицировать файлы в процессе их сохранения на резервный носитель: для клонов эти изменения невидимы.

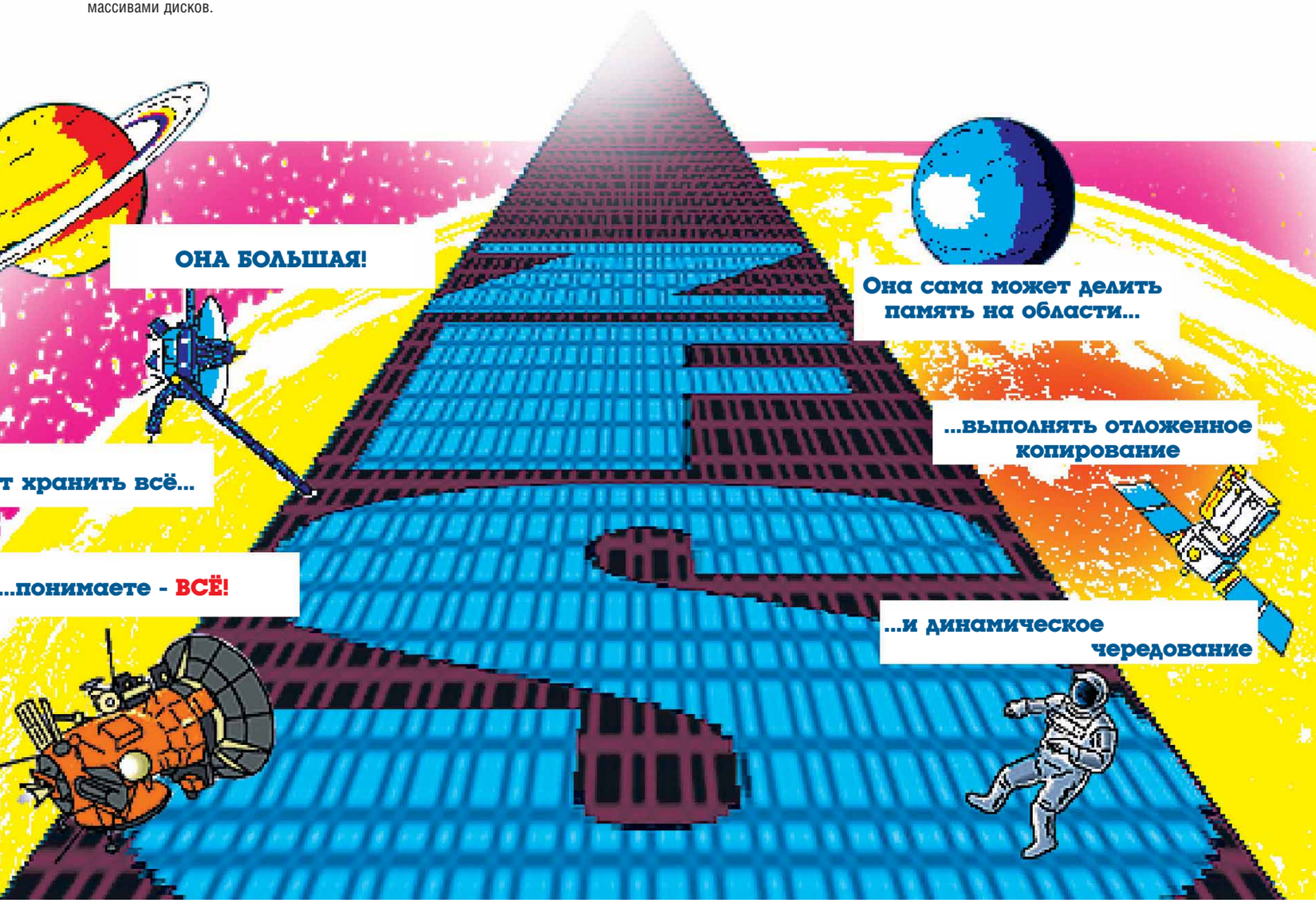
Вы были правы – от всего этого «крыша едет»!
ZFS изменила даже базовую концепцию жестких дисков: появилось понятие «область памяти». На Linux истинная гибкость файловой системы может быть достигнута лишь при использовании менеджера томов типа LVM, но ZFS это ни к чему, так как можно включить в область памяти столько угодно устройств, а затем на основе созданной области построить собственно файловую систему.

А что такое «чередование»?
В действительности все устройства, включаемые в область памяти, являются виртуальными и могут быть как физическими приводами, так и RAID-массивами дисков.

Круто, но я не о том спрашиваю. Можно ли чередовать данные в областях памяти ZFS?
Конечно, можно. А динамическим чередование называется потому, что в область памяти можно свободно добавлять новые виртуальные устройства, при этом ZFS автоматически изменит параметры чередования, чтобы извлечь максимум пользы из увеличившегося числа опознанных устройств. Размер области памяти вырастет, а все операции чтения/записи будут распределены между новыми модулями так, чтобы нагрузить их равномерно.

Хитро придумано. А можно ли установить ZFS на мою Linux-машину?
Это не так просто. Лицензия ZFS несовместима с лицензией ядра Linux, поэтому скомпилировать ее в ядро прямо сейчас невозможно. Но существует проект (см. http://zfs-on-fuse.blogspot.com) по созданию FUSE-интерфейса для ZFS, и, по типу проекта NTFS-3g, ZFS сможет работать в пользовательском пространстве. На этом сайте можно более подробно прочесть о том, почему ZFS такая крутая, а также о том, что предпринимается для ее переноса на OS X и FreeBSD.

Скукотища... А нет ли более занимательного способа узнать больше?
А я-то надеялся, что вы об этом не спросите. Но, коли так, посмотрите видеоролик Sun Germany о том, какая ZFS крутая. Найти его можно по адресу http://video.google.com/videoplay?docid=8100808442979626078. Только не проклинайте меня, если в результате не поймете, на что потратили 10 минут своей жизни – это «странность» совершенно нового уровня. Ролик наглядно демонстрирует, насколько ZFS отказоустойчива: показано, как она бесперебойно справляется с областью памяти, состоящей из 12 USB-накопителей, причем все они отключаются и подключаются в случайном порядке! LXF



Мнение: Джереми Эллисон

«Я не могу менять законы физики!»

Джереми Эллисон

Джереми – ведущий разработчик команды *Samba*, сейчас работает на Google, в Сан-Хосе, Калифорния.

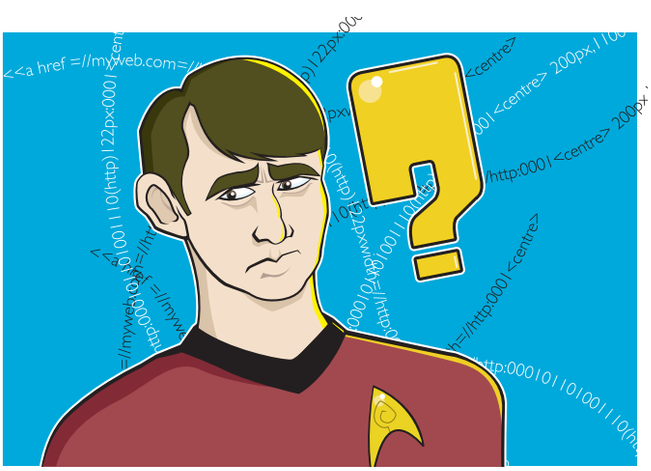
Инженер Скотт всегда был моим любимым персонажем в Стар Трек. Пускай капитан Кирк и очаровывал всех зеленокожих дам подряд, но если ему было что-то нужно от его потрясающего корабля, он волей-неволей обращался к Скотту. И если Скотт сообщал ему, что кристаллы дилитиума истощились, капитан плакал, как дитя, но в итоге все равно вынужден был ждать, пока Скотт решит проблему.

Мне всегда казалось, что Скотту следовало бы почаще говорить капитану «нет», особенно когда требования Кирка шли вразрез с законами физики. Однако инженеры не слишком склонны отказывать начальству, и бедняге Скотту приходилось творить чудеса неделю за неделей, пока даже я, будучи ребенком, не заподозрил, что этими свершениями он в большей мере обязан сценаристам, чем своему глубокому пониманию Вселенной.

Попытки заставить технические средства защиты цифровых авторских прав работать в реальном мире сродни попыткам заставить инженеров совершать вместо нормальной работы чудеса в стиле Стар Трека. DRM (Digital rights management, технические средства защиты цифровых авторских прав) не может работать в принципе. Как вы знаете, DRM – это попытка контролировать копирование цифрового файла, или, иногда, ограничивать количество сделанных копий. Потребителю никогда не раскрывают его истинной сущности: это просто ограничение на их действия с тем, за что они уже заплатили. DRM обычно определяют как «чудесную новую технологию, которая поможет защитить вашу медицинскую карту от злоумышленников». А правда заключается в том, что оно этого сделать не в состоянии.

> «Хватит и одного хакера, чтобы сделать тайное явным: ни в одной версии DRM секретов нет.»

О DRM часто говорят в связи с шифрованием, которое и в самом деле является весьма полезной технологией, действительно способной защитить ваши медицинские данные от злодеев. Однако шифрование и DRM – это разные вещи. В шифровании десятилетия научных изысканий уходят на создание изощренных алгоритмов и методов, вроде криптографии с открытым ключом, на которой основана практически вся интернет-коммерция, чтобы любители совать нос в трафик получали только нечто шумоподобное. Смысл любого шифра в том, что ключ хранится в секрете. Он обязан быть секретным, поскольку именно он является гарантом сохранения тайны данных, которыми владеют две стороны.



А теперь рассмотрим DRM. DRM применяется к цифровым данным одной стороной – как правило, это владелец музыки или фильма – и шифрует данные, требующие защиты, с помощью алгоритма шифрования и ключа. Другая сторона – потребитель – получает зашифрованные данные, информацию об использованном алгоритме и копию ключа к шифру. Все это обязательно передается потребителю, иначе он не сможет прослушать или посмотреть данные, которые приобрел. Однако предполагается, что DRM ограничивает действия потребителя над этими данными. И каким образом это можно сделать? С помощью чудо-кристаллов дилитиума?

Трубить об обеспечении полной безопасности этого процесса от людей, которым выдана вся информация – это все равно, что очертить мелом круг на асфальте и объявить его безопасным банковским хранилищем: пусть, мол, клиенты складывают внутри круга деньги, «уважая установленные границы». Медиаиндустрия стремится продать то, что она считает ценной информацией, без всяких средств запрещения доступа к этой информации. Эта бизнес-модель никогда не будет работать.

Недавнее появление скрытых ключей шифрования для HD-DVD – превосходный пример этого фантастического инженерного подхода. Последнюю «супербезопасную» систему DRM уже дважды взломали. Взлом не грозит только тем системам DRM, которые лишены маломальски интересного содержимого. Совершенно не важно, сколь изощренны уровни программ, добавляемых к системе, нужен всего один человек с отладчиком или логическим анализатором – и он прорвется сквозь них, и обнаружит тайную истину: что ни в одной версии DRM на самом деле нет никаких секретов.

Инженеры понимают, что DRM не сможет работать. Увы, как и Скотт, когда капитан Кирк со своего мостика требовал от него невозможного, они не в состоянии прекратить выпуск все более изощренных версий все той же взломанной системы. Компании продолжают создавать системы DRM и продавать их информационной индустрии.

Во время недавнего разговора в Google Кори Доктороу [Cory Doctorow], активист движения за электронную свободу, автор произведений в жанре научной фантастики и создатель аггрегатора новостей для компьютерщиков Boing Boing, сказал, что требование, чтобы инженеры создавали систему DRM – показатель отказа потребителя воспринимать действительность рационально. Они по-прежнему верят в бизнес-модель гномиков в штанишках из Саус Парка: Шаг 1 – создать систему DRM. Шаг 2 – ??? Шаг 3 – Прибыль! «Печально, – особо отметил он, – что правительство США, похоже, тоже живет в мире фантастики, как и – хоть и в меньшей степени – правительство Великобритании.»

На данный момент в мире более всего вводят всех в заблуждение попытки правительства США заставить другие страны принять так называемые «права на интеллектуальную собственность» в американском стиле. Они основаны на теории, что США и Великобритания способны уступить свою производственную базу Индии или Китаю и при этом удерживать ведущие позиции в мире, контролируя информацию, используемую для разработки продукции или продажи цифрового контента разбогатевшим потребителям этих стран. Утверждается, что в будущем цифровые биты будет легче контролировать и сложнее копировать. А ведь вся история технологий доказывает обратное!

Верить в бизнес-модель DRM – то же самое, что вступить в службу безопасности Звездного Флота, нацепить красную рубашку и отправиться на новую неизученную планету вместе с Кирком, Споком и МакКой. Кто-то сумеет оттуда вернуться, но сотрудники службы безопасности – вряд ли. Скотту, как истинному инженеру, хватит здравого смысла, чтобы оставаться на корабле. LXF

LINUX FORMAT

Учебники

Наши эксперты помогут вам с любым приложением Linux

ЕВГЕНИЙ БАЛДИН
Начинал с Агатов. Когда-то даже знал, что такое Робик.

GNU/Linux не бесплатный — это просто правильный выбор?

Мир достаточно велик, чтобы удовлетворить нужды любого человека, но слишком мал, чтобы удовлетворить людскую жадность.

Мохандас Карамчанд «Махатма» Ганди

Если нужно кого-то убедить в том, что на тот или иной компьютер/класс/отдел следует поставить GNU/Linux, то не надо налегать на бесплатность. У GNU/Linux есть масса других более важных достоинств и одно из них – это стабильность.

Как заставить человека покупать новую версию офисного продукта, если старая его всем устраивает? Продукт следует сделать несовместимым с предыдущей версией. Этот трюк пройдёт в случае, только если есть жёсткая привязка к закрытым форматам. Куда денешься, если начинают приходить тесты с тем же расширением, но открывающиеся немного не так? Но почему обычные пользователи должны платить за чужую жадность?

Если не будет специальных законодательных запретов, то через тридцать-сорок лет фактически все программы общего применения заведомо будут «гнутыми». Кодовая база свободного программного обеспечения только увеличивается и никуда не теряется, то есть «становиться хуже» и «исчезать вообще» она гарантировано не будет. Это стратегический выбор на перспективу.

Можно сказать, что пользователям не нужно задумываться о перспективе, но ИТ-специалистам самое время. Ведь в случае стабильности наработанные в прошлом навыки не будут пропадать и можно будет совершенствоваться, а не переучиваться.

Да, и пусть следующий ваш компьютер будет с предустановленным GNU/Linux.

E.M.Baldin@inp.nsk.su

В этом выпуске...

56 Редактируем фильмы онлайн
YouTube хорош для просмотра, а известно ли вам, что можно стать самому себе режиссером, не имея ничего, кроме браузера? Энди Ченнел знает, как.

60 Bash для начинающих
Поначалу командная строка выглядит устрашающе, но Рейчел Пороберт уже готова зажечь спасительный факел и осветить темные углы *Bash*.

64 Учимся работать в Vim
Вниманию тех, кто по-прежнему сидит в *Nano*! Только сегодня д-р Крис Браун проводит эксклюзивный мастер-класс по прокачке навыков редактирования текста с переходом на *Vim*.

68 Сетевое программирование
Да, Пол Хадсон не считает Mono номером один для написания сетевых приложений – но это не мешает ему написать свой собственный чат!

72 Эксперт по apt-get
Ubuntu не сводится к Synaptic! В вашем распоряжении – вся мощь системы управления пакетами APT и Пол Хадсон, жаждущий поделиться своими знаниями!

76 Матрица переходит на Java

Мало кто задумывается о том, что машинам Матрицы ежесекундно приходилось обрабатывать более 6,5 млрд. запросов, поступающих от сидящего в капсулах благодарного человечества. **Александр Бабаев** подозревает, что Архитектор просто использовал NIO.

80 Печатаем на ризографе

С иллюстрациями все – пришла пора испачкать бумагу. **Александр Чернышов** покажет, как сделать это с минимальным вредом для окружающей среды.

82 Finita la comedia

На какие только ухищрения не идут создатели современных блокбастеров, лишь бы заставить вас просмотреть все титры до конца. **Петр Семилетов** покажет, как реализовать их смелые замыслы в Cinelerra.

84 Радиус обслуживания

Сегодня мы поговорим о беспроводных сетях, но не в том разрезе, о котором вы подумали. **Андрей Боровский** покажет, как настроить безопасность WPA Enterprise, используя только компьютер с установленным Linux.

Совет месяца: Вески аргументы

Мы не можем научить вас выигрывать каждый спор, но можем показать вам, как овладеть другим типом аргументов: аргументами командной строки. Вы наверняка уже знаете как передавать программе несколько параметров, например:

```
someprog file1 file2 file3
```

Это хорошо и прекрасно, пока вы вводите команды вручную, но что, если вы хотите использовать список, созданный другой программой? Скажем, например, вы хотите открыть в *GIMP* десять самых больших изображений в каталоге. Вы можете выполнить *ls -l*, что бы найти их, и затем муторно вводить каждое имя файла по типу **gimp file1 file2**... Или вы можете использовать магию *xargs* (она есть в каждом дистрибутиве).

```
ls -S --color=never | head -n10 | xargs gimp
```

xargs собирает сложные операторы командной строки, используя списков файлов, тем самым избавляя вас от необходимости набирать их вручную. Вы просто передаете *xargs* список файлов и он превращает их в строку **file1 file2 file3**..., которую вы видели ранее. Понимаете? Взгляните на пример выше. Сначала мы используем *ls*, чтобы вывести список файлов в текущем каталоге (расцветка отключается, поскольку она добавляет ненужные нам управляющие символы). Затем, используя старый добрый канал (|), мы передаем результат утилите *head*, которая отбирает первые 10 строк.

Итак, сейчас у нас есть список из десяти файлов. Но как же нам сказать *GIMP*, что мы хотим открыть их? Мы не можем просто передать имена через канал, поскольку *GIMP* воспримет их как данные изображения. Здесь-то и приходит на помощь *xargs*: он гарантирует, что следующая программа получит список аргументов командной строки, а не информацию на стандартном вводе (stdin). Приведенный выше пример буквально говорит: «Возьми первые 10 строк, сгенерированные head и передай их *GIMP* в стандартном формате **file1 file2 file3**...

Вы можете сделать с *xargs* многое; все, что она от вас ожидает – список файлов, разделенный символами перевода строки или пробелами. Например, вот так можно удалить все файлы старше 10 дней (используйте с осторожностью)

```
find . -mtime +10 | xargs rm
```

Jumpcut: Видеоредактор

Редактирование видео в Linux наконец стало простым даже для самых зеленых новичков... если с web-браузером все в порядке. Встречайте **Энди Ченнела**.

Наш эксперт

Энди Ченнел
Энди делает свои первые шаги в Linux уже шесть лет, а технологиями интересуется еще со времен Dragon 32.

Работа с видео в Linux может казаться сложным делом. Не потому, что нет программ для захвата и редактирования, а потому, что им не хватает Drag-n-Drop-простоты таких приложений, как *iMovie*, *Windows Movie Maker* или *Adobe Premiere Elements*. И хотя *Cinelerra* и менее объемный *Lives* невероятно мощны, но они требуют значительного времени и усилий, чтобы выжать из них максимум.

К счастью, новый вид сетевых приложений, похоже, заполняет пробел с перетаскиванием на рынке Linux. На данном уроке мы кратко рассмотрим видеозахват при помощи *Kino* и видеокамеры с FireWire и загрузим наш клип в Интернет для разрезания на фрагменты при помощи *Jumpcut*. Затем мы отшлифуем наш фильм, подготовив к показу друзьям и родственникам. Более подробная информация о захвате видео в Linux есть в LXF65, в сетевом архиве **Linux Format** (**www.linuxformat.co.uk/pdfs/LXF65.tut_begin.pdf**).

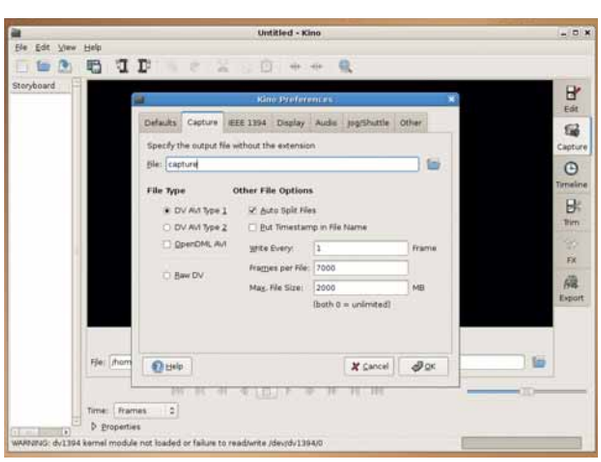
Kino тоже очень неплохо умеет редактировать, хотя мы здесь и будем использовать его лишь для захвата – другими словами, для переноса фильма на ПК в корректном формате. Если ваши амбиции простираются далее пятиминутного прикола, рассылаемого по сети, вам обязательно следует его рассмотреть. Приложение имеет версию 1.0 и должно быть доступно в менеджере пакетов вашего дистрибутива. Если это не ваш случай, подключайтесь к сети и заходите на сайт **www.kinodv.org**, чтобы найти подходящую ссылку для загрузки.

Для захвата видео вам понадобится цифровая камера с портом FireWire и карта FireWire-интерфейса для вашего Linux-ПК – и, естественно, некий готовый отснятый материал. FireWire иногда называют IEEE1394 или iLink на камерах Sony DV, и он хорошо поддерживается в дистрибутивах Linux – это настолько хорошо поддерживаемый стандарт, что при подключении камеры в гнездо FireWire вам не придется ничего настраивать. Если ПО не распознает ваше устройство, откройте диалог Edit > Preferences [Правка > Настройки] и посмотрите на вкладке IEEE 1394. Скорее всего, вы (как пользователь) не имеете прав доступа к файлу **/dev/1394**, но это можно изменить, запустив апплет управления группами вашего дистрибутива (в Ubuntu это System > Administration > Users And Groups [Система > Администрирование > Пользователи и группы]) и записавшись в члены группы Video.

Мягкий переход

Перед тем, как начать захват, вновь откройте диалог Preferences [Настройки] и выберите вкладку Capture [Захват]. Здесь вы можете указать расположение и имя каждого видеофайла, который будет записан (Kino добавит число к каждой новой секции), выбрать формат сохранения и установить кое-какие другие параметры. Мы собираемся выбрать AVI тип 1, потому что это стандартный файл с интегрированным звуковым потоком. Тип 2 записывает звук отдельно, что хорошо для высококлассного редактирования. Одно из ограничений выбранного нами сетевого редактора – это предел в 100 МБ для загружаемого файла, так что, возможно, вы пожелаете установить Max File Size [максимальный размер файла] в 99 МБ, чтобы при сохранении позднее производилась автоматическая разбивка. Также выберите Autosplit [Авторазбиение], позволив *Kino* распознавать отдельные части вашего фильма и сохранять их в отдельных файлах.

Выберите вкладку Capture [Захват], затем щелкните на кнопке AV/C внизу экрана, чтобы разрешить управление подключенной камерой. Управление лентопротяжным механизмом *Kino* теперь можно использовать для перемещения по вашей пленке в поиске гениальных кадров. Когда будете готовы к захвату, нажмите кнопку Capture [Захват] – каме-



› Kino имеет немало опций захвата, и, к счастью, сервис Jumpcut отлично работает с AVI-файлами.

Месяц назад: Мы научились публиковать файлы в Сети через F-Spot.

в сети

ра начнет воспроизведение, а компьютер будет записывать поток и помещать его в отдельные AVI-файлы в указанном ранее месте. Вам остается просто сидеть и смотреть.

Насчет выгрузки вашего фильма в Интернет придется брать волю в кулак. Мои пять минут про крысу в саду вылились в четыре файла общим весом почти 31 МБ, что при стандартном высокоскоростном соединении (нечего и говорить, что не стоит даже пытаться делать это по телефонной линии) требует 30 минут на выгрузку. К счастью, поскольку это процесс выгрузки, то он не очень помешает вашему блужданию в Интернете, поскольку оно в основном связано с загрузкой.

Вперед, на выгрузку…

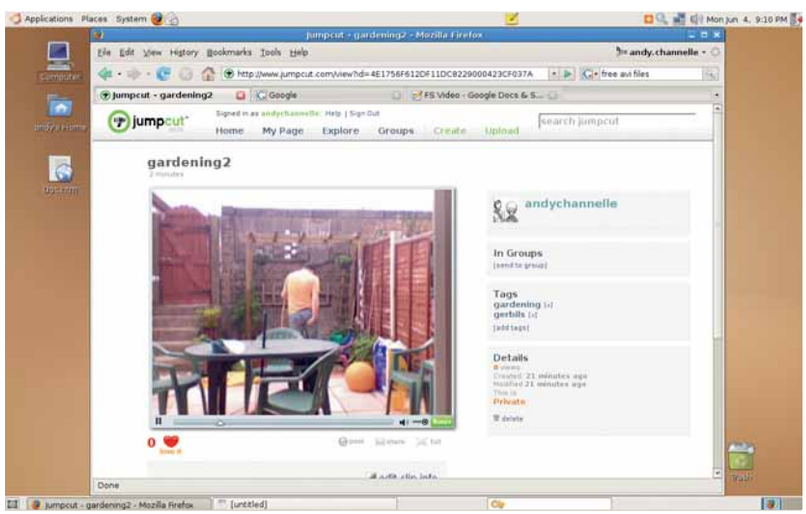
Чтобы начать загрузку, перейдите на web-сайт *Jumpcut* **www.jumpcut.com** (имеются и другие сетевые службы редактирования видео) и зарегистрируйтесь или войдите в готовую учетную запись. Щелкните на ссылке Upload [Выгрузка], она находится у правого края панели меню в верхней части окна (не панели меню браузера!). Поскольку это первое наше использование системы, она нам скажет, что у нас пока нет фильмов для редактирования.

В разделе Movies [Фильмы] щелкните на ссылке с текстом Upload Your Own Video Clips And Photos [Выгрузить ваши собственные видеоклипы и фотографии]. Запустится Flash-загрузчик (заранее позаботьтесь об установке Flash 9); теперь вы можете выбрать один или несколько файлов для выгрузки. При нажатии массивной кнопки Upload [Выгрузка], загрузчик откроет стандартный диалог выбора файлов, и вы сможете выбрать несколько файлов клавишами Shift или Ctrl. Выделив фрагменты, которые хотите выгрузить, нажмите Upload [Выгрузить] и заварите чай или полейте цветы.

Та же процедура выгрузки применима для добавления звука или фотографий в вашу коллекцию. Только убедитесь, что вы не используете музыку, на которую у вас нет прав, если хотите сделать ваши творения достоянием общественности.

После начала выгрузки вам дается возможность добавить к вашим клипам права доступа и метки-теги. Первые варьируются в пределах от от общих [public] до личных [private], а последние используются именно для того, чего вы и ожидали: для назначения клипам категорий.

› Выгрузка файлов съедает немало времени. Вам вряд ли захочется выгружать «Русский ковчег».



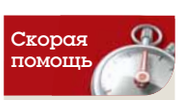
Это особенно полезно, если вы предоставляете свои клипы в общее пользование, разрешив просмотр и переделки. Если, например, вы засняли огнеглотателя и присвоили клипу метку «Пожиратель огня», другие пользователи смогут найти его без многочасового перематывания роликов. По окончании загрузки, файл отобразится в окне предварительного просмотра, посреди окна браузера. Для просмотра ваших файлов нажмите Play [Воспроизведение].

Следующая стадия – более тщательный предварительный просмотр клипов. На главной странице *Jumpcut* вы должны увидеть последние добавленные файлы под заголовком Clips [Клипы]. Щелкните на одном из них для предварительного просмотра – помните, что клип после этого будет загружаться с сайта *Jumpcut*, так что, в зависимости от скорости вашего соединения, может пройти несколько секунд, прежде чем начнется воспроизведение.

Этот экран содержит некоторую интересную информацию. В самом верху, под заголовком, отображается время клипа. Внизу справа – информация о самом клипе, включая количество просмотров, метки и права доступа. Имеется также маленькая иконка удаления. Сразу под окном предпросмотра находятся кнопки Post [Разместить], Share [Опубликовать] и Full [На весь экран]. Кнопка Post предоставляет возможность добавить клип к web-страницам или любым другим популярным сайтам социальных сетей, включая Blogger, MySpace и Facebook. Кнопка Share позволяет отправить ссылку на этот клип по списку электронных адресов, чтобы пригласить на просмотр вашего клипа лишь избранных, а не всех подряд. Кнопка Full просто показывает ваш клип во все окно браузера.

Под этими опциями находится кнопка Edit Info, позволяющая редактировать заголовок и «историю» клипа, а ниже находится строка ввода Comments [Комментарии], в которой вы можете добавить заметки к любому клипу.

› Linux-хакер чешет… з-з, затылок. В этом окне редактируются настройки для отдельных клипов и добавляются заметки или комментарии.



Если вы хотите редактировать видео в сети, но у вас нет собственного материала, почему бы не переделать чужой контент, используя *Jumpcut*? Это тоже круто.

Первый дубль

Что ни говори, захват и выгрузка видео – утомительное занятие. А вот редактирование превосходно. Можно не только преобразовывать реальность, используя монтаж в стиле Эйзенштейна, но и добавлять дурацкие эффекты и эффектные переходы между сценами – даже в web-приложении типа *Jumpcut*. Но для начала давайте добавим к нашему проекту несколько звуковых и видеофайлов. Нажмите кнопку »

» Create [Создать] у верхнего края для перемещения к окну Make A Movie [Создать фильм]. Мы выберем в Start A New Edit Session [Начать новую сессию] пункт Open The Editor [Открыть редактор]. Для несложного монтажа можно использовать мастера *Jumpcut*, но мы хотим большего контроля над результатом.

После небольшого ожидания, пока загружаются магические Flash-компоненты, перед вами появится главный интерфейс *Jumpcut*. Он соответствует стандартам обычного настольного приложения: имеется панель меню, панель предварительного просмотра, настройки и, внизу главного окна, временная шкала. Вначале все должно быть пустым, но мы исправим это, перейдя на вкладку Clip и нажав кнопку Add [Добавить]. На вкладке Your Media [Ваши записи] в выпадающем списке выберите All Of Your Clips And Photos [Все ваши клипы и фотографии], чтобы заполнить библиотеку всем, что вы выгрузили на сайт. Коллекцию можно также пополнить, выбрав вкладку Upload [Выгрузка] и просмотрев файлы на вашей машине.

По сути, процесс редактирования заключается в переносе клипов из библиотеки на монтажный стол, и это первое, что следует сделать. Создайте черновую сборку путем переноса файлов на стол. Над шкалой времени, справа от элементов управления воспроизведением, находится видеоползунок. Он используется для мгновенного перемещения к клипу или месту в клипе и очень пригодится позднее, когда мы начнем наводить лоск. Здесь также возможно сделать некоторые неразрушающие изменения (то есть файлы на самом деле меняться не будут) в синхронизации сцен, щелкая и перенося указатели в конец или начало, тем самым передвигая их слева направо внизу под окном предпросмотра. Стрелка, направленная вправо – точка «входа», а влево – точка «выхода». Выберите каждый из клипов в порядке следования на монтажном столе для редактирования их точек входа/выхода или, на техноязе, для выравнивания клипов.

Мы можем стать более аккуратными с нашим выравниванием, выбрав клип на монтажном столе, а затем щелкнув на вкладке Clip ниже окна библиотеки. В разделе Duration [Длительность] вы увидите точки входа/выхода, выраженные в числах (секундах), и сможете точно выровнять клипы, изменив значения. Помните: это недеструктивно, так что экспериментируйте вволю.

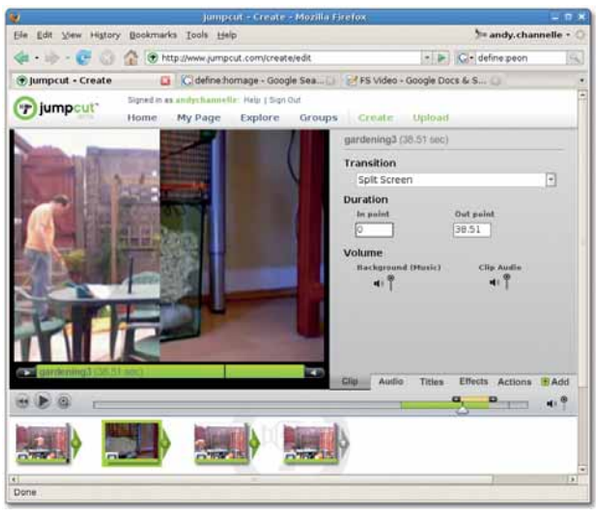
Скорая помощь

Jumpcut поддерживает множество видеоформатов, включая 3GP, то есть вы можете отправлять ролики по электронной почте прямо с сотового телефона.

Переходный период

Итак, у нас есть клипы, они находятся в правильном порядке и отлично выровнены. Пора добавить несколько переходов. Должен отметить, что так можно либо внести динамику, либо сделать полную безвкусицу. Только вы можете решить, что есть что, экспериментируя и развлекаясь.

Выберите клип на монтажном столе и щелкните на вкладке Clip. Интересующее нас находится в выпадающем списке Transitions [Переходы]. *Jumpcut* имеет небольшой, но полезный набор доступных переходов, включая несколько любопытных вариантов вроде Dissolve [Растворение] (когда один клип, затемняясь, переходит в другой) и откровенно ужасные штуки вроде Countdown [Обратный отсчет] и CrissCross Horizontal [горизонтальное скрещивание], которые, однако, довольно популярны. После добавления перехода можете отредактировать его длительность, используя уже знакомые стрелочки выравнивания, которые появятся в месте соединения двух клипов. Для увеличения длительности перехода сделайте так, чтобы он чуть больше наезжал на оба клипа; чтобы переход был быстрее, сделайте его меньше.

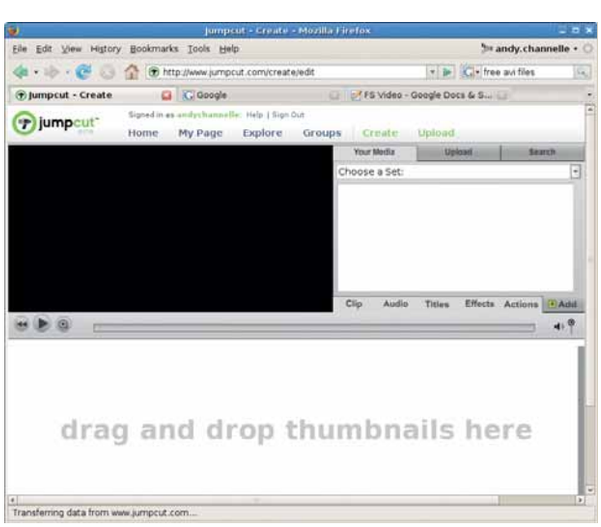


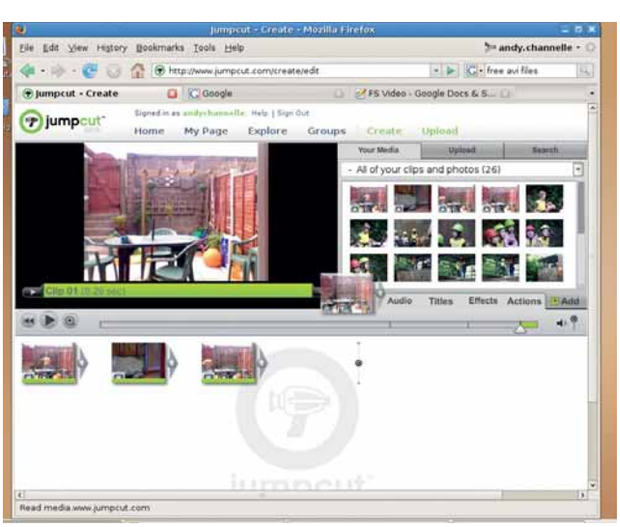
› Переходы Jumpcut бывают удивительно эффектны. И они действительно запускаются в браузере.

Внизу окна имеется несколько инструментов для удаления, дублирования и разрезания клипов. Разрезание особенно полезно, если вы хотите разбить клип и вставить в его середину другой. Для разрезания клипа, установите указатель воспроизведения точно в то место, где следует произвести разрез, затем нажмите кнопку Slice [Разрезать]; теперь вы сможете передвигать обе части независимо друг от друга.

Справа от инструментов редактирования клипов вы найдете несколько действий над файлами, включая кнопки Save [Сохранить] и Preview [Предпросмотр]. Если вы добавили массу эффектов (читайте о них далее), полезно сделать предпросмотр, чтобы выяснить, не слишком ли они замедляют показ.

Настраивая переход, помните, что он привязывается к концу выбранного клипа и переносится вместе с ним; он также останется в конце клипа, если вы позднее выровняете точку выхода. Однако переходы не деструктивны, и вы сможете удалить неудачный вариант, если, взглянув на свою первую поделку, разочаруетесь в своем чувстве прекрасного.





› Интерфейс пользователя выглядит пустоватым, пока вы не приметесь добавлять свои записи.

При всей простоте, переходы *Jumpcut* могут быть очень эффективными. Например, медленное растворение может использоваться для затемнения и появления людей (точнее, голубых полицейских коробчонок) на сцене; с помощью пары дублей и длинного перехода с разделением экрана можно заставить человека разговаривать с самим собой на другой половинке экрана. Хитрость, как и во всяком творчестве, в переборе вариантов до тех пор, пока вы не получите желаемый результат.

Эффективность эффектов

Спецэффекты в видео, практически без исключений, полностью бессмысленны, и их магия, вероятно, полиняет еще до завершения первого показа вашего опуса. Их бесполезность превосходит лишь удовольствие забавляться с ними.

Для начала, выберите клип на монтажном столе, затем нажмите кнопку Effects [Эффекты]. Справа от окна предпросмотра в прокручиваемом списке появятся доступные эффекты. Просмотр эффекта заключается лишь в его выборе и нажатии кнопки Play ниже окна предпросмотра. Когда вы добавляете эффект, при отрисовке клипа монтажный стол слегка пульсирует, но отрисовка происходит гораздо быстрее, чем вы могли бы ожидать, и результата своих действий ждать недолго. Не нравится? Щелкните на другом эффекте и просмотрите его – свобода в работе с ними потрясающая – и если хотите объединить эффекты, просто щелкните на вкладке Clip.

Если вы твердо решили сохранить эффект в вашем клипе (а надо ли?), нажмите кнопку Add [Добавить], и он добавится в клип. Мощь эффектов, как и других опций редактирования, в том, что они абсолютно не деструктивны, так что вы можете опробовать их и даже опубликовать результат без реального изменения исходников – что идеально для запуска видеопродукции с открытым кодом. Более того, можно наложить эффекты слоями один на другой, превратив клип в черно-белый, добавить наложение камеры и снабдить все это телевизионной рамкой. Но не горячитесь, поскольку уже при трех примененных эффектах появляется недостаток: просмотр начнет тормозить.

Добавляя или выбирая эффект, заметьте, что появляется новая полоска выше указателя воспроизведения с уже знакомыми левой и правой стрелочками в конце. Вы можете изменить длительность эффектов, передвигая их.

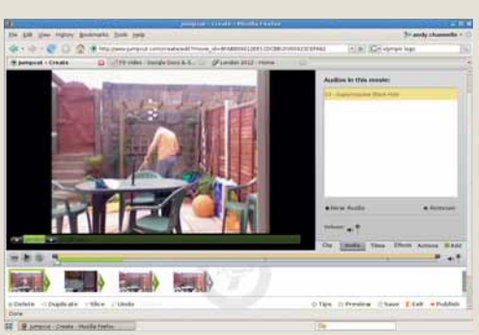
Еще один, особый класс эффектов доступен на вкладке Titles [Титры]. Здесь можно добавить некий поясняющий текст: имена, даты или некоторые детали. Он работает так же, как и другие эффекты, за исключением того, что один раз выбрав желаемый стиль титров, вы можете вводить текст, который будет отображаться в текстовой области под списком. И вновь, все остается полностью редактируемым: прежде чем остановиться на одном конкретном стиле, можно перепробовать остальные (и это здорово). Можете изменить время отображения титров перемещением левой/правой стрелочек в полосе выше указателя воспроизведения, или их расположение, щелкнув на них и перетащив в другое место на полосе видеоползунка; можете даже перенести титры в другие клипы. Назовите меня провинциалом, но от стиля Animated Lines [Живые строки] у меня сердце поет!

Итак, мы выравнивали и разрезали видео, добавляли искусные эффекты, оттачивали наши титры и озвучили все это крутым роком. Блокбастер *Садоводство и крысы* готов к премьерному показу, так что нажмите кнопку Publish и будьте прокляты. Можете отослать друзьям ссылки на этот продукт, сделать его общим или сохранить лишь для персонального использования. Если вам нужна локальная копия фильма (или вообще любое Flash-видео), используйте *Firefox Video Downloader* с https://addons.mozilla.org/firefox/2390/. LXF

Воспроизводим

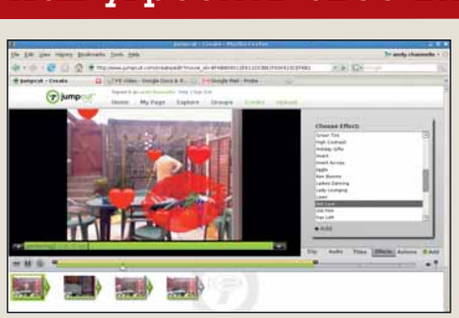
Вы можете пожелать добавить звуковую дорожку к вашему фильму, и это вполне возможно в *Jumpcut*. Щелкните на вкладке Audio, выберите один из звуковых файлов и нажмите Add [Добавить]. (Если вы еще не добавили ни одного звукового файла, то нажмите кнопку Add и или выберите какой-нибудь с вашего диска, или прочешите библиотеку *Jumpcut*). При добавлении в проект звукового файла над указателем воспроизведения появится оранжевая полоса, которую можно подвергнуть выравниванию, как и видеоклипы. Около нижней кромки окна Audio находится регулятор громкости клипа. Добавив звук, нажмите кнопку Play [Воспроизведение] внизу окна предпросмотра, чтобы получить представление о текущем состоянии проекта.

Приятно, что вам не нужно ограничивать себя одним звуковым файлом для сопровождения вашего творения: жмите кнопку New Audio [Новый звук] для добавления второго, третьего или четвертого файла. Оранжевая линия и регулятор громкости соответствуют звуковому фрагменту, выбранному в настоящий момент, так что весьма просто создать плотные звуковые миксы без нужды запускать какой-нибудь звуковой редактор. Обычно это бывает полезно, когда у вас есть некие закадровые или окружающие шумы (то есть звук, записанный вместе с видео), музыкальный саундтрек и голос диктора в одном клипе. Правда, звуковые настройки будут применены ко всему файлу, поэтому невозможно наложить голоса на плавающий саундтрек (делающийся тише, когда говорит диктор) без обращения к чему-то вроде *Audacity*. Но вы можете изменять громкость отдельных клипов по сравнению со вторым звуковым треком при помощи ползунка на вкладке Clip.

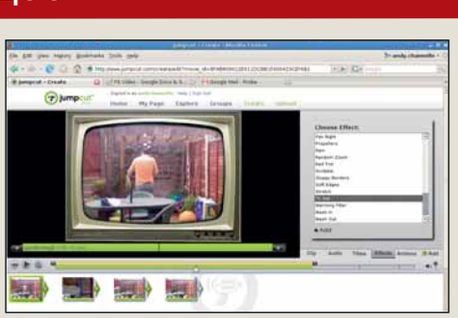


› Исправьте длительность аудиоклипа, используя стрелки влево/вправо над видеоползунком.

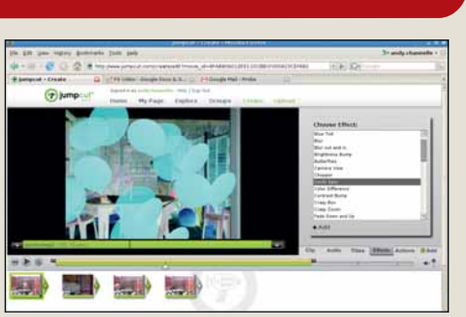
Как угробить свое видео



› Net Love – превосходный эффект для моего видео с крысой в саду. Почувствуйте любовь.



› Если вы всегда мечтали попасть в телевизор, вот вам идеальная возможность.



› Вспомните славные деньки Top of the Pops (с добавлением садоводства) при помощи фильтра Circle Spin!

» **Через месяц** Приводим дела в порядок при помощи Linux.

Bash: Работаем с

Часть 1: Хотите немного порезвиться в командной строке, но не знаете, с чего начать? **Рэйчел Проберт** покажет несколько базовых команд, которые пригодятся новичкам.

Мы начинаем новую серию статей о работе в командной строке с использованием *Bash*. Собираетесь ли вы сделать хобби своей профессией или просто хотите разобраться в своей домашней сети, вам потребуется овладеть приемами работы в командной строке. Интерфейс командной строки (CLI, command line interface) предоставляет пользователю большие возможности по сравнению с графическим интерфейсом (GUI, graphical user interface). Обычно он еще и быстрее. Освоив несколько команд, которые вы будете использовать ежедневно, вы сэкономите массу времени и сил.

Представьте, что вам нужно срочно найти написанный несколько недель назад отчет. Имя файла вы точно не помните, но знаете, что отчет был связан с компанией ABC. Сразу после того, как вы его найдете, отчет нужно отправить коллеге. Обычно это занимает несколько минут. Но *Bash* и немного волшебства (см. команды ниже) помогут сделать все за десять секунд:

```
grep -r -i -H CompanyABC /home/rjprobert/work
cat filename | mail colleague@work.com
```

Наверное, пока эти команды ни о чем вам не говорят, но к концу этой серии статей вы будете точно знать, что к чему, и уверенно использовать их каждый день. Однако, начнем с азов. Сегодня мы посмотрим, как работают команды управления доступом root, su и sudo и когда ими нужно пользоваться. Мы рассмотрим создание новой учетной записи, а также научимся перемещаться по каталогам и просматривать их содержимое.

Наш эксперт

Рэйчел Проберт последние семь лет консультирует начинающих бизнесменов и разрабатывает для них web-приложения..

Часть 1: Root, su и sudo

Итак, что же такое «доступ с привилегиями администратора (root)»? Каждый, у кого есть такой доступ, словно владеет ключами от города. Права root позволяют вам делать все что угодно, в том числе, изменять важные системные файлы. Один неверный шаг – и систему нужно переустанавливать заново. Например, команда *rm -rf /* в мгновение ока удалит все файлы, находящиеся на компьютере. Поэтому большинство администраторов используют этот режим доступа только в тех случаях, когда это совершенно необходимо. Я люблю проводить аналогию между работой в режиме администратора и девизом одного из моих любимых супергероев: «Большая сила – большая ответственность». В общем, не используйте доступ с привилегиями администратора без необходимости, а когда используете, будьте максимально осторожны. Если вы не знаете, от имени какого пользователя вы работаете, это очень просто проверить. Откройте окно терминала (Приложения > Дополнительные > Терминал). Появится маленький черный экран с белой стрелкой приглашения (>). Посмотрите на последний символ перед мигающим курсором. Знак доллара '$' означает, что вы зашли в систему как обыкновенный пользователь, а символ решетки '#' – как супергерой.

Итак, что же нам остается? Что делать, если мы не должны заходить в систему как пользователь root? Ответ таков: нужно работать под обычным пользователем и использовать команды *su* или *sudo* для решения задач администрирования. Каждый раз, когда для решения задачи требуются привилегии администратора системы, вы можете ввести команду *su* (сокращение от Substitute User – заместить пользователя). После этого у вас запросят пароль администратора, вознаграждением за который послужит выполнение команды. Вы по-прежнему можете нанести системе серьезный ущерб, вы можете делать что угодно и где угодно.

Лучшим вариантом будет настройка и использование команды *sudo*, с помощью которой можно ограничить тот ущерб, который вы можете причинить. Как администратор системы, вы должны определить, что может делать каждый пользователь и где он может это делать. Для выполнения команды предварите ее командой *sudo*. После запроса пароля система выполнит только ту команду, которую вы имеете право дать. Добавить новых пользователей и изменить привилегии существующих можно в файле **/etc/sudoers**. Для изменения файла подойдет любой текстовый редактор, но гораздо лучше использовать команду *visudo*. Это одна из тех задач, для решения которой вам потребуются права администратора системы.

Man-страница, в которой вы найдете все необходимое для добавления пользователей в список *sudoers*, выдается командой:

```
man sudoers
```

В конечном счете, вы и только вы выбираете средства, с помощью которых риск падения системы от опечаток и непродуманных действий будет сведен к минимуму. Но пока вы помните мантру «root не для рутинной работы», ошибок быть не должно. А сейчас давайте перейдем к повседневным задачам, которые вы можете решить с помощью командной строки.

пользователями

LINUX FORMAT На DVD

Часть 2: Создание новой учетной записи

Для начала давайте познакомимся с некоторыми командами. Мы создаем новую учетную запись для себя (или кого-то другого), т.е. для обыкновенного пользователя, для решения повседневных задач, таких как отправка почты и работа с документами. Для того, чтобы добавить нового пользователя в систему, я буду использовать команду *adduser* вместе с именем пользователя «rjprobert». Это простой и удобный способ создания учетной записи. У вас будет запрошен пароль пользователя и некоторая другая информация.

Введите следующую команду, заменив моего пользователя своим. Запомните, что эта операция потребует привилегий администратора, поэтому вам придется использовать команду *su* или *sudo* в зависимости от того, как вы хотите использовать свою силу.

```
adduser rjprobert
```

Когда вы создаете нового пользователя, для него создается подкаталог в каталоге **/home**. Мы может проверить, что новый пользователь появился, с помощью команды:

```
ls /home
```

Команда *ls /каталог* просто выводит его содержимое. Таким образом, мы должны увидеть каталог нового пользователя, которого только что добавили.

> «Пока вы помните мантру «root не для рутинной работы», ошибок быть не должно»

С помощью различных опций команды *ls* можно получить дополнительную информацию. Сама команда *ls* выводит только имена файлов в текущем каталоге. Вы можете также попробовать *ls d**, чтобы вывести список всех файлов, имена которых начинаются с '*d*' и так далее. Попробуйте команду *ls -l* которая выведет список файлов, отсортированных в алфавитном порядке. Информация о каждом файле должна выглядеть примерно так:

```
-rw-r--r-- 1 rjprobert users 132422 June 05 11:22 mypic.jpg
```

Эта информация состоит из прав доступа к файлу (см. *Первые шаги* LXF91), числа ссылок на файл, имени пользователя и группы, которым он принадлежит, размера в байтах, даты и времени его создания и, наконец, имени файла.

У команды ls есть более 50 чувствительных к регистру опций. Это означает, что *ls -g* и *ls -G* вернут разные результаты: первая команда выведет подробный список файлов (как и *ls -l*), но без информации о владельце файла, в то время как вторая – подробный список файлов без информации о группе. Полный список опций команды *ls* можно найти на ее man-странице.

Если вы захотите сменить каталог и оказаться в каталоге **/home**, просто введите команду:

```
cd /home
```

Итак, мы только что использовали команду cd. Это не так трудно! Давайте посмотрим, как еще мы можем применять ее для перемещения по другим каталогам. Команда *cd /[имя каталога]* приведет вас в один из основных каталогов системы. Чтобы подняться на уровень выше, используйте *cd ..* Чтобы зайти в подкаталог, вам потребуется команда *cd [имя подкаталога]*, а просто *cd* вернет вас в домашний каталог.

Надеюсь, вы поняли, насколько гибким и мощным инструментом может быть *Bash*. В качестве домашней работы попробуйте побродить по вашей системе с помощью команды *cd* и узнайте, где что живет. Совет: если вы потерялись и не можете вспомнить, где находитесь, вас всегда выручит команда *pwd*. Она просто выводит полный путь текущего каталога. Итак, вам есть чем заняться до следующего месяца! LXF

Полезный совет

Bash завершает имена файлов при нажатии клавиши Tab. Наберите *cd* и первую букву(ы) названия каталога, и клавиша Tab завершит его имя для вас.

Что такое Bash?

Когда вы используете командную строку Linux, всегда запущена программа, которую называют оболочкой. Оболочка – это программа, с помощью которой вы взаимодействуете с системой. Когда вы набираете команды в строке терминала, именно оболочка считывает, обрабатывает их и передает операционной системе. Есть много различных программ-оболочек, но в большинстве систем Linux (как и в Mac OS X) по умолчанию используется *Bash*. Это название представляет собой акроним фразы «Bourne-again shell», игра слов с «рожденная снова» (born again) оболочка Bourne shell – очень важная оболочка, написанная давно Стефеном Борном [Stephen Bourne].

Помощь в командной строке

Вместе с Linux поставляется впечатляющая система интерактивных справочных руководств, часто называемых man-страницами. Они содержат исчерпывающую информацию как о ядре, так и об остальных утилитах.

Набрав *man любая_команда*, вы получите полную информацию по использованию команды. Руководства разбиты на разделы, которые включают в себя:

1. Пользовательские команды
2. Системные вызовы
3. Вызовы библиотечных функций
4. Специальные файлы
5. Форматы файлов
6. Игры
7. Прочее
8. Команды администрирования

Команды Linux, вызовы функций и файлы часто снабжаются номером раздела, который указывается в круглых скобках после имени команды. Например, *passwd(1)* означает, что информацию можно получить с помощью команды *man 1 passwd*. Как новички, обычно мы будем использовать только первый раздел.

» **Через месяц** Инструментарий продвинутого пользователя: *grep* и каналы

Evolution: Быть

Надоели наклейки повсюду – памятки о намеченных встречах? **Пол Хадсон** покажет более разумный – и более в стиле Gnome – способ организовать свою жизнь...

Вам нужны:

- Сумасшедший стиль жизни
- *Evolution* от Gnome
- Место для публикации календаря

Вам нужны:

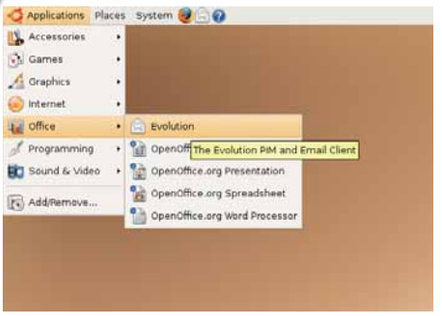
- Мгновенная организация
- Быстрые ссылки на все события
- Совместное ведение календаря

Если ваше представление об организации жизни – это погоня за кем-то другим, значительным, кто намного лучше помнит о датах и о записях в ежедневнике, то пора привлекать к делу компьютер: пусть выполняет за вас трудную работу. Ведь это его обязанность, разве нет?

Оказывается, есть стандартный формат файлов для календарной информации, известный как ICS. Он уже используется для всех типов календарей, и сотни их доступны на www.icalshare.com. Но ни один из них не поможет организовать лично вашу жизнь, так что следующие 10 минут вы будете изучать, как настроить собственный календарь и стать организованнее с помощью превосходного персонального органайзера Gnome под названием *Evolution*. Являясь частью рабочего стола Gnome, *Evolution* обеспечивает отличную функциональность календаря, и вы сможете любоваться ею каждый раз, когда кликнете по панели даты/времени в Gnome. Это означает, что любое событие, добавленное вами в календарь *Evolution*, будет появляться там автоматически.

Evolution также обладает встроенной функцией публикации календаря, и вы сможете показывать свое расписание другим, независимо от того, каким календарем они пользуются. Хотя на данном уроке мы пользуемся Gnome и *Evolution*, на самом деле многие программы умеют публиковать календарь в сети, и мы добавили кое-какую полезную информацию, чтобы помочь вам попробовать все это в своем любимом приложении.

Evolution, календарь и Gnome

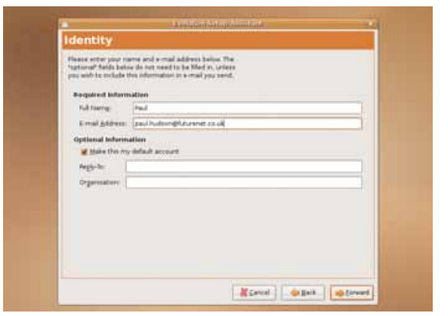


1 Запуск *Evolution*

Чтобы запустить *Evolution*, зайдите в Приложения > Офис > Evolution. В большинстве дистрибутивов Gnome *Evolution* устанавливается по умолчанию, хотя мы не слишком часто используем его для электронной почты, потому что web-почта сейчас уже весьма неплоха.

2 Настройка почтовой учетной записи

Если вы еще не запускали *Evolution*, вас попросят настроить почтовую учетную запись. Это немало раздражает, особенно если вы не собираетесь использовать *Evolution* для проверки почты! Но чтобы безболезненно избежать вопросов, следуйте нашим указаниям. Нажмите здесь Далее.

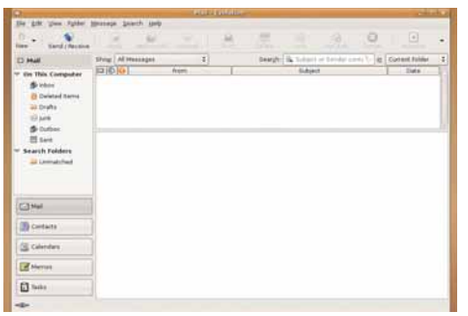


3 Добавка подробностей

На этом экране вам надо указать имя и адрес электронной почты – можно даже фиктивный, это неважно, мы все равно не будем им пользоваться. Впишите какую-нибудь ерунду, и затем дважды кликните Далее.

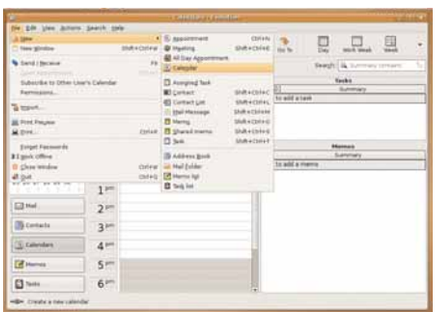
4 Отправка почты (или нет)

Тут *Evolution* попросит вас выбрать способ отправки почты, но нам это ни к чему, так что опять введите в строку SMTP какую-нибудь ерунду; или воспользуйтесь более быстрым выходом: измените тип сервера на Sendmail, трижды кликните Далее, и затем Применить.



5 Настройки *Evolution*

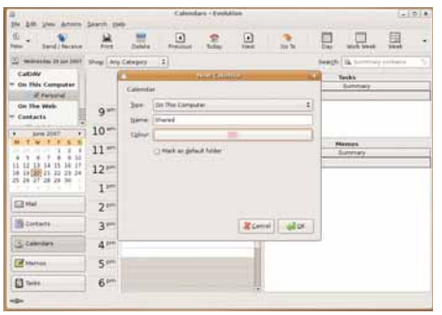
Наконец-то *Evolution* загружается – и выглядит довольно пустынно, поскольку мы изо всех сил заставляли его игнорировать почту. Не бойтесь, вам не придется снова брести через все эти диалоги, потому что с этого момента и далее *Evolution* будет помнить все ваши настройки.



6 Создание календаря

Внизу слева в окне кликните Calendars, чтобы переключить *Evolution* на работу с календарем. *Evolution* по умолчанию создает «личный» календарь для вас, он пуст, но мы создадим новый календарь с открытым доступом. Выберите Файл > Создать > Календарь.

порядку в дневнике



7 Открытие доступа к календарю

Измените тип календаря на On This Computer и дайте ему имя ‘Shared’. Поскольку доступ к этому календарю открыт, я выделил свой красным цветом, чтобы помнить: не вносить сюда сугубо личную информацию!

8 Публикация

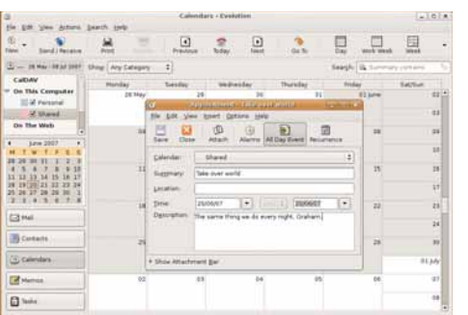
Теперь выберите Правка > Параметры, потом Календарь и задачи, затем выберите вкладку Публикация календарей. По умолчанию, ни один из ваших календарей не будет опубликован в Интернете – вам надо указать *Evolution*, какие из них вы хотите опубликовать.

9 Выбор места размещения

Нажмите кнопку Добавить, и перед вами появится диалоговое окно. *Evolution* позволяет выбрать, где будет размещен ваш календарь – то есть там, где он будет доступен всем. В нашем примере мы выбрали WebDAV, но для большинства самым простым будет Ресурс Windows.

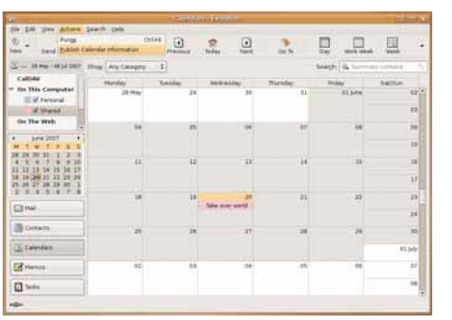
10 Ввод имени файла

Введите расположение сервера и имя файла в строки под разделом Тип сервиса. Если вы используете Ресурс Windows, убедитесь, что имя файла включает в себя имя разделяемого ресурса. Если при размещении потребуется ввести имя пользователя и пароль, введите их.



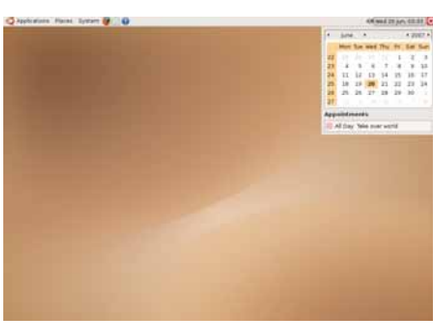
11 Добавление события

Нажмите OK, и вы снова вернетесь в календарь. Теперь выберите Вид > Текущий Вид > Как месяц и добавьте событие, просто дважды щелкнув по определенному дню. Нажмите Сохранить, и в вашем календаре появится запись.



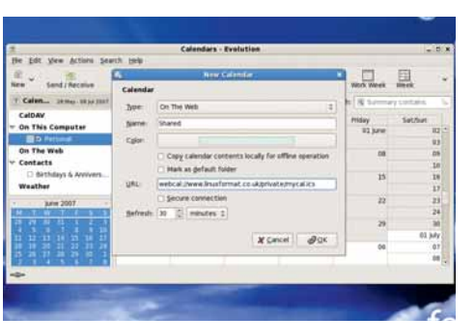
12 Внесение изменений

Теперь перейдем в Действия > Публиковать сведения календаря, чтобы внести изменения в те данные, которые открыты для общего доступа. Можно настроить *Evolution* так, чтобы он автоматически публиковал изменения через регулярные промежутки времени, но будет проще, если вы сделаете это сами.



13 Просмотр внесенных записей

Все записи, которые вы добавляете в свой календарь, автоматически появляются на рабочем столе Gnome – щелкните по панели время/дата, и увидите календарь со списком всех событий на день. На рисунке – наше напоминание о событиях.



14 Ввод пути

Теперь переходим на другой компьютер – тот, на котором вы хотите тоже видеть свой календарь. В *Evolution* на другой машине возвращаемся на экран Новый календарь и на этот раз выбираем тип On The Web и вводим путь к вашему календарю с открытым доступом.

15 Использование iCal

Если вы пользуетесь *iCal* на Mac, он тоже работает – выберите Calendar > Subscribe, затем введите полный URL вашего календаря с открытым доступом (включая http://, если он опубликован в сети). Затем можете выбрать имя и частоту обновления. Легко! LXF

Vim: Введение

Почтенный *Vim* – любимый редактор многих Linux-профессионалов. Он имеет репутацию трудного в освоении, но стоит трудов, уверяет **д-р Крис Браун**.

Наш эксперт

Д-р Крис Браун независимый инструктор по Linux, имеет степень доктора по физике элементарных частиц и сертификаты Novell CLP и Red Hat RHCE. Написал книгу о SUSE для издательства O'Reilly.

Текстовых редакторов в Linux намного больше, чем вы можете себе представить. Похоже, хотя бы по одному создали каждый человек и его собака. Продукты от собак обычно норовят обнюхивать гнезда на задней панели компьютера, и их, вероятно, лучше игнорировать, но и без них остается много – *emacs*, *kedit*, *gedit*, *kwrite*, *kate*, *mousepad*, *leafpad*, *fte*, *joe*, *jed*, *nano*, *pico*… список можно продолжить. Но большинство профессионалов Unix и Linux выбирает их общего дедушку, *Vi*, и его собрата помоложе, *Vim*. (*Vim* происходит от «Vi improved» [улучшенный *Vi*] – это *Vi* , переписанный в 1991 г. Брэмом Муленаром [Bram Moolenar]. Исходный *Vi* написал Билл Джой [Bill Joy] в далеком 1976 г. На данном уроке используется *Vim*, хотя основной набор команд для *Vi* и *Vim* одинаков.) Одна из причин популярности *Vim* – эффективность редактирования даже при отсутствии графического интерфейса: например, если вы перевели машину в однопользовательский режим для профилактики, или это сервер, где не установлен рабочий стол.

Vim не самый простой редактор для изучения – его практически невозможно понять без чтения хоть какой-то документации. Не ждите также мгновенной отдачи. Как преподаватель Linux при переходе к разделу о *Vim* я вижу в глазах студентов ужас. Один парень сказал мне, что если нужно будет редактировать файлы Unix, он скорее скопирует их на дискету и отредактирует в Windows, чем изучит *Vim*. Это отвращение к *Vim*'у позорно, потому что, изучив его, вы будете редактировать текст намного быстрее, чем ранее. Так что стисните зубы, прочитайте учебник и станьте *Vim*-гением.

Произносите правильно

Vi следует произносить как «ви-ай». Вы опозоритесь перед знатоками Unix, если произнесете это как «вай». И уж точно не следует говорить «шесть», как предложил один мой коллега, вообразивший, что он происходит от римлян...

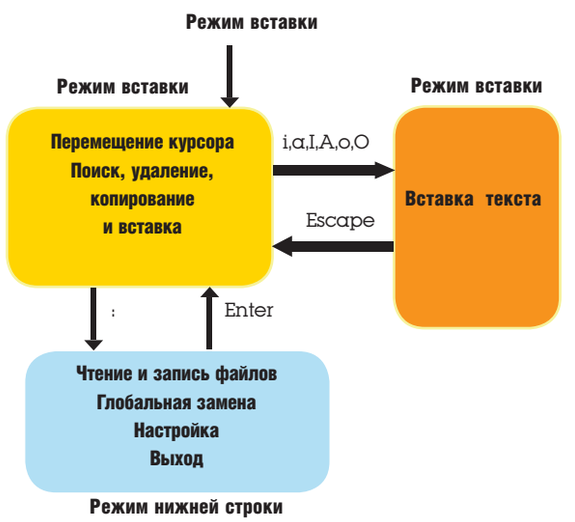


Рис. 1. Режимы *Vim* и их функции.

Обычно *Vim* запускается с именем файла в качестве аргумента. Файл открывается для редактирования, если он существует – а если нет, то создается. Так что начнем с команды

```
vi gpl.txt
```

Если вы хотите работать с тем же файлом, загрузите себе его копию, используя

```
wget www.gnu.org/licenses/gpl.txt
```

Одна из причин трудности изучения *Vim* и *Vi* – их режимы работы (некоторые скажут «перебор режимов!»). Другими словами, результат ввода символов зависит от режима, в котором вы находитесь. Например, в командном режиме ввод w вызывает переход на одно слово вперед, а в режиме вставки просто в текст вставляется 'w'. Режимы и переходы между ними показаны на **Рис. 1**.

Командный режим

Стартуя, *Vim* входит в командный режим. Большинство команд – это отдельные символы; наиболее полезные из них приведены в **Таблице 1**.

Классические команды перемещения курсора в *Vi* и *Vim* – h, j, k и l, выполняющие перемещение на один символ влево, вниз, вверх и вправо. Мнемоники тут никакой: они выбраны потому, что это базовые клавиши при печати вслепую, и всегда под руками. (В *Vim* используются также клавиши-стрелки, но поскольку к ним нужно еще передвинуть руку, получается медленнее.) Большинство команд перемещения курсора могут иметь числовой префикс для повтора команды; так, 5j перемещает на пять строк вниз, а 3w – на три слова вперед. В команде G префикс указывает номер строки: например, 23G перенесет вас на 23-ю строку.

В ОСНОВЫ

Поиск в тексте при помощи / и ? – это наискорейший способ перемещения по нему. Искомый текст может быть (ограниченным) регулярным выражением, например, /[Ss]oftware будет искать **Software** и **software**. Если вы нашли не тот экземпляр нужной строки, продолжайте вводить n для поиска следующего.

Режим вставки

Для добавления текста вы должны перейти в режим вставки. Простейшие команды для этого – i, производящая вставку перед текущей позицией курсора, и a, вставляющая после него. Команды I и A – укрупненные версии команд i и a; они производят вставку до и после текущей строки соответственно. Команды o и O создают пустую строку ниже или выше текущей, а затем переходят в режим вставки.

Будучи в режиме вставки, вы можете вставлять что угодно: от одного символа до нескольких строк. Однако в режиме вставки нельзя перемещать курсор или давать любые другие команды редактирования – для этого придется вернуться в командный режим (хотя есть версии, где можно заставить использовать клавиши-стрелки для перемещения курсора даже в режиме вставки).

Имеется лишь один способ вернуться из режима вставки в командный: нажать Esc. Заведите привычку нажимать Esc почаще.

Удаление и изменение текста

Команда x удаляет один символ. Она имеет префикс повтора, так что 6x удаляет шесть символов. Команда d более мощная, и должна сопровождаться командами перемещения курсора, указывающими объем удаляемого текста. Вот как она работает: d, в комбинации с перемещениями курсора, удаляет текст между его начальным и конечным положением. Например, dw удаляет слово, d3w удаляет три слова, d$ удаляет все до конца строки, d} удаляет все до конца абзаца, а dG удаляет все до конца файла. Имеется также специальная версия команды d – dd, которая удаляет всю строку. Префикс повтора работает и здесь, так что 3dd удалит три строки.

Обождем минутку... не являются ли команды поиска (/ и ?) просто видом перемещения курсора? Можете ли вы использовать их с командой d? Оказывается, можете. Например, d/PUBLIC удалит все до слова PUBLIC (не включая его). Более элегантный пример, d/^[0-9] удалит все до первой строки, начинающейся с цифры. (Когда я преподаю *Vim*, обычно в этом месте враждебность уступает неохотному признанию силы набора команд.)

Команда c (change [изменить]) работает почти как команда d, за тем исключением, что она к тому же переводит редактор в режим вставки. Так что c3w можно рассматривать как указание «изменить три слова»; и так далее.

Команда u отменяет последние сделанные изменения. *Vim* поддерживает многоуровневую отмену – каждая u отменяет одно изменение. В исходном *Vi* она работает не так. Если вы ввели две u, то вторая отменит действие первой!

Режим нижней строки

Режим нижней строки используется для всего, но команда нижней строки, без которой не обойтись – это w для записи отредактированного текста в исходный файл и q для выхода. Из командного режима можно перейти в режим нижней строки, набрав :, так что :wq –стандартный способ записать ваш файл и выйти из редактора. Если вы хотите выйти из редактора без сохранения, используйте :q!. Попозже мы рассмотрим другие команды нижней строки. »

Таблица 1: Базовые команды

Перемещение курсора	
h, j, k, l	Перемещение курсора на один символ: влево, вниз, вверх, вправо
w	Вперед на одно слово
b	Назад на одно слово
0	Начало строки (цифра ноль)
$	Конец строки
(	Начало предложения
)	Конец предложения
{	Начало абзаца
}	Конец абзаца
G	Конец файла
%	Перейти к соответствующей парной скобке того же уровня
Поиск	
/string	Первое вхождение строки (поиск вперед)
?string	Первое вхождение строки (поиск назад)
n	Следующее вхождение (поиск продолжается в том же направлении)
N	Предыдущее вхождение (поиск продолжается в обратном направлении)
Переход в режим вставки	
i	Вставить текст перед курсором
a	Вставить текст после курсора
I	Вставить в начало строки (заглавная 'i')
A	Добавить в конец строки
o	Создать новую строку ниже текущей
O	Создать новую строку выше текущей
Редактирование текста	
x	Удалить один символ
d	Удалить (и скопировать в буфер удаления)
y	Выдернуть (скопировать в буфер удаления)
p	Положить (вставить) содержимое буфера удаления
c	Изменить
.	Повторить последнее изменение (но в текущей позиции курсора)
u	Отменить последнее изменение
J	Объединить текущую строку с предыдущей
<	Уменьшить отступ
>	Увеличить отступ

Скорая помощь

Рассматривайте свой текст не как двумерный массив символов, а как последовательность слов и предложений. Тогда перемещаться в *Vim* будет легче.

Скорая помощь

Если вы запускаете *Vim* в окне терминала, то не думайте, что меню окна терминала может что-то делать в *Vim*. Это не так! Можно вставлять текст в *Vim* при помощи меню окна терминала Edit > Paste [Правка > Вставить], но *Vim* не отличит его от вводимого с клавиатуры, так что сначала убедитесь, что вы в режиме вставки, иначе вставляемый текст будет рассматриваться как команда!

» Вы узнали достаточно, чтобы рулить в *Vim* по крайней мере на второй скорости, но если вы хотите уметь развить суперскорость и выполнить разворот на месте, читайте дальше...

Операции вырезания и вставки

Vim поддерживает операции вырезания и вставки, используя буфер удалений. Текст, удаляемый вами по команде d, сохраняется в этом буфере и может быть вставлен обратно в новой позиции курсора командой p (положить). Например, переместить текущий абзац в конец файла можно так:

1. Перейти к началу текущего абзаца при помощи {
2. Удалить абзац командой d}
3. Перейти в конец файла командой G
4. Вставить удаленный абзац обратно при помощи p

(Сравните-ка эти несколько команд с числом действий мышью, необходимым для выполнения того же действия в редакторе с управлением мышью.)

Копирование и вставка выполняются практически так же, за тем исключением, что вы используете команду y вместо d. Команда y «выдергивает» текст – то есть помещает его в буфер удаления, но на самом деле не удаляет. Как и d, она действует совместно с командами перемещения курсора, так что y2) скопирует два предложения, а 4yy скопирует четыре строки. Например, для дублирования текущей строки, наберите yyp, чтобы скопировать строку и вставить копию ниже.

Скорая помощь

Воспринимайте командный режим как стандартный, а режимы вставки и нижней строки как временные, используемые по мере надобности. Не воспринимайте режим вставки как основной. Если вы не уверены, в каком режиме находитесь – нажмите ESC.

Становится круто

Если одного буфера удаления вам мало, то вам будет приятно узнать, что их 26, с именами от ‘a’ до ‘z’. Двойные кавычки используются для указания имени буфера. Например, “t4yy скопирует четыре строки в буфер с именем ‘t’, а “tp вставит содержимое буфера ‘t’. Содержимое именованных буферов сохраняется, пока вы работаете в редакторе или пока вы его не перезапишете.

Vim выполняет также «глобальные» подстановки. Допустим, вы решили переделать текст лицензии GPL с американского английского на британский, заменив все слова ‘license’ на ‘licence’. Имеется два способа: ручной и автоматический. Ручной выглядит так:

1. Перемещаемся к началу файла командой 1G
2. Ищем первое вхождение ‘license’ при помощи /license
3. Меняем его на ‘licence’, используя cwlicence<ESC>
4. Ищем следующее вхождение при помощи n
5. Если вы хотите и его заменить, наберите . (точка). Эта команда повторяет последнее сделанное изменение (в данном случае, команду cw) в текущей позиции курсора
6. Продолжайте так по всему файлу, используя n и . для повторения поиска и замены.

Если вы просто хотите заменить ‘license’ на ‘licence’ во всем файле насквозь, без просмотра отдельных изменений, можете использовать команду подстановки нижней строки *Vim*. Для данного примера она выглядит так:

```
:1,$s/license/licence/g
```

Рассмотрим ее поподробнее:

1. Команда : переносит нас в режим нижней строки.
2. Запись 1,$ указывает диапазон номеров строк. 1 – первая строка, а $ означает последнюю. Если мне необходимо сделать замену только в первых 50-ти строках, я должен буду использовать здесь 1,50.
3. Запись s/license/licence – простой пример команды подстановки
4. g в конце означает ‘global’; то есть, если указанный текст содержится в строке больше одного раза, то заменить все экземпляры.

Vim выдаст число сделанных замен и номера измененных строк.

Опции

Для настройки *Vi* и *Vim* могут использоваться десятки изменяемых опций. Те, что я счел наиболее полезными, приведены в **Таблице 2**, но их намного больше. Для изменения опции следует использовать команду нижней строки set. Некоторые опции являются логическими – они принимают значения только on [включено] или off [выключено]. Например, для изменения опции автоотступа следует ввести :set autoindent (: служит для перехода в режим нижней строки, помните?). Для выключения этой опции введите :set noautoindent. Имена многих опций к тому же имеют сокращения. Например, автоотступ можно укоротить до ‘ai’, так что :set ai и :set noai тоже сработают. Другие опции имеют строковые или числовые значения. Опция shiftwidth, например, указывает количество символьных позиций, используемых для отступа. Ясно, что это числовое значение, устанавливаемое как-то вроде :set shiftwidth=4.

Можно определить «постоянные» значения опций (они будут читаться при каждом запуске *Vim*), поместив их в файл **~/.vimrc**. Например, строка этого файла вида set sw=2 ai запустит *Vim* в режиме с автоотступом с величиной смещения 2. (Заметим, что в файле **.vimrc** нет : перед командой set).

Фильтрация

Среди всех возможностей *Vim*, одна из моих любимых – это возможность обрабатывать текстовые буферы (или любую их часть) при помощи внешнего фильтра Linux.

Фильтрация использует команду !, которая, как и команда d, требует сопутствующих перемещений курсора. Текст между текущей позицией курсора и тем местом, куда он переместится, следуя командам, пропускается через выбранные вами внешние команды, и результат их действия вновь вставляется в текстовый буфер вместо начально-

Таблица 2: Опции Vim

Опция	Аббревиатура	Тип	Что делает
number	nu	bool	Включает нумерацию строк
showmode	smd	bool	Заставляет Vim отображать название режима, используя метку в нижней строке
cindent	cin	bool	Включает отступы в стиле C
ignorecase	ic	bool	Игнорировать регистр при поиске
magic	-	bool	Учитывать регистр при поиске
wrapmargin	wm	number	Номер позиций относительно правого края экрана, где Vim вставит перенос строки в режиме вставки
autoindent	ai	bool	Включение автоотступа
shiftwidth	sw	number	Число символов отступа
matchpairs	mps	string	Список парных символов, используемых для функции ‘showmatch’ и для команды %. Набор по умолчанию: matchpairs=(:),{:},[:]
wrapscan	ws	bool	Осуществление поиска по тексту с конца файла к началу

› *Vim* **в не-графическом терминале. Мыши здесь места нет!**

го содержимого. При использовании команды ! курсор помещается в нижнюю строку, чтобы дать место для ввода команд фильтрации. Более детально о фильтрах я поговорю потом, но пара примеров поможет понять идею.

Фильтр cat, с опцией -n, добавит к выводу номера строк. Последовательность команд 1G!Gcat -n выполнит следующее:

1. Перемещение в начало файла (1G)
2. Выбор всего файла для фильтрации (!G)
3. Указание cat -n в качестве фильтра. (Эта часть команды будет выведена в нижней строке.)

Заметьте, что вставка номеров строк этим способом и в самом деле добавляет их к буферу редактирования. Это не то же самое, что включение нумерации строк при помощи опции :set number, которая просто велит *Vim* нумеровать строки на экране.

Другой пример: последовательность команд 1G!Gwc запустит wc для всего файла, чтобы выдать нам количество символов, слов и строк. Прочитав это, используйте команду u для отмены изменений и замены вывода команды wc исходным текстом.

Наш третий пример фильтрации использует двойную форму команды ! (!!), для работы с полными строками, и использует *grep* и подходящее регулярное выражение для фильтрации пустых строк. Вот команда: 10!!grep -v '^$'. А вот как она работает:

1. 10!! выбирает десять строк для фильтрации.
2. Команда grep -v запускает *grep* для вывода строк, не соответствующих регулярному выражению.
3. Регулярное выражение ^$ означает «начало строки, за которым сразу же идет конец строки» – то есть соответствует пустым строкам.

Функции для программистов

В *Vi* и *Vim* имеется множество функций специально для программистов. Во-первых, имеется режим автоотступа, активируемый при помощи опции :set ai. При включенном автоотступе, по мере ввода текста, *Vim* автоматически отодвигает новую строку в соответствии с предыдущей. Если необходимо, вы можете увеличить отступ при помощи ^T или уменьшить – ^D.

Для исправления отступов существующих строк, используйте команды > и <. Они работают так же, как и команда d, и требуют указания перемещений курсора, так что >} увеличит отступ до конца абзаца (то есть до следующей пустой строки). Однако эти команды в основном используются в сдвоенном виде, например, 4>> сделает отступ у четырех строк.

По умолчанию, отступ равняется восьми позициям – вероятно, это многовато. Вы можете изменить размер отступа (*Vim* называет это 'shiftwidth') командой вроде :set sw=4.

Еще одна дружественная к программистам функция – способ управления различными типами скобок. Если курсор находится на скобке одного из типов (,), {, }, [или], то ввод % переместит курсор к соответствующей скобке того же уровня вложенности. Конечно, вы можете использовать % для перемещения курсора так же, как и команды d, c или y, и если вы редактируете исходный код на языке C, в котором используются фигурные скобки для ограничения блоков, вы можете удалить блок очень просто, поместив курсор к открытой скобке и набрав d%.

Кроме того, если вы установите опцию showmatch командой :set showmatch, Vi немедля переместит курсор к соответствующей открывающей скобке каждый раз, когда вы вставите закрывающую скобку, такую как), } или]. Исходный *Vi* не умеет этого делать с угловыми скобками – < и >, и *Vim*, по умолчанию, тоже; однако вы можете его заставить, добавив <:> к строке опции matchpairs по команде :set matchpairs+=<:> – это пригодится, если вы редактируете HTML или XML-документы.

Кроме того, имеется подсветка синтаксиса, которую лично я не очень люблю, хоть и знаю, что многие программисты считают ее полезной. Исходный *Vi* ее не поддерживает, поскольку терминалы тех дней могли отображать только один цвет (обычно зеленый). В *Vim* она включается при помощи команды нижней строки :syntax enable. *Vim* автоматически распознает тип файла и загружает соответствующую цветовую схему.

Мы только начали...

Имеется еще много команд и опций *Vim*, для которых у нас просто нет места. Если вы читаете это, будучи экспертом *Vi*, и я пропустил вашу любимую функцию, пожалуйста, простите меня! Втиснуть в четыре страницы все, что пользователи *Vi* считают заслуживающим внимания, просто невозможно, так что рассматривайте это как беглое введение. Есть расширенная документация по *Vim* на www.vim.org/htmldoc/ и PDF-версия книги Стива Оллина [Steve Oualline] о *Vim* на www.truth.sk/vim/vimbook-OPL.pdf. И заметьте, что даже я готов признать: нужно быть большим энтузиастом редактора, чтобы для ознакомления с его работой пуститься читать 572-страничную книгу... LXF

Скорая помощь

При использовании *Vim* отложите мышь подальше и установите обе руки над клавиатурой. Все операции редактирования *Vim* у вас под руками в «базовых клавишах» слепого метода печати, и для профессиональных наборщиков это реальное преимущество.

Немного истории

Дизайн *Vi* станет более понятен, если рассматривать его в контексте устройств, существовавших в 1976 г., когда Билл Джой написал его. Типичным терминалом того времени был Lear-Siegler ADM3A, машина на основе ЭЛТ-монитора (перейдите по ссылке www.vintage-computer.com/otheritems.shtml – и увидите такую). ADM3A имел полную QWERTY-клавиатуру, но без функциональных клавиш и стрелок, имеющихся на современной клавиатуре. Дизайн *Vi* отражает эти ограничения: все перемещения курсора и операции редактирования могут быть выполнены без их использования.

ADM3A был одним из первых терминалов, имевших адресуемый курсор (то есть вы могли отсылать ему специальные последовательности ASCII-символов для определения места появления на экране следующего символа). Сейчас мы принимаем это как должное, но тогда это была новация, и она вызвала всплеск активности: люди стали писать настоящие «экранные редакторы» для замены прежних строковых редакторов. Такие терминалы подключались к компьютерам через последовательный порт – либо напрямую, либо при помощи телефонных соединений по модему, который тогда предоставлял скорость обмена 300 или (у везунчиков) 1,200 bps. В некоторых экранных редакторах после удаления строки и вставки пары слов вы могли за время обновления экрана успеть не то что выпить, а наварить пива. Одним из преимуществ *Vi* была эффективность обновления экрана, наряду с возможностью управлять большим числом терминалов, используя базу данных termcap. В последних версиях *Vi* и *Vim* все еще есть переменная среды TERM, определяющая тип терминала.

Mono: Создание

Если звенят ваши чувства паука, значит, где-то происходит преступление. А если звякнула ваша струнка Mono – значит, **Пол Хадсон** припас вам новый проект...

Наш эксперт

Пол Хадсон полагает, что Mono лучшая вещь со времен мультфильма Pinky and the Brain и сейчас поддерживает два проекта на основе Mono на SourceForge.

Хотя счета за телефон норовят доказать обратное, я не сильно люблю болтать с трубкой. Вот женщины – у тех непонятная сверхспособность разговаривать по телефону одновременно с утюжкой или просмотром телевизора, а я обнаружил, что могу сконцентрироваться на том, что говорят люди, только если их понимаю. Поэтому я предпочитают мгновенные сообщения – через программки для чата, позволяющие быстро обмениваться репликами с друзьями (или с людьми, которые нашли где-то в сети ваш электронный адрес и чувствуют себя одинокими). При помощи IM-клиентов я могу читать сообщения, ставить их в мою мысленную очередь для пакетной обработки и заняться своими делами в ожидании ответа. Более того, мне не нужно скучливо дожидаться, пока оппонент печатает – ведь я прочту все сообщение, когда он нажмет кнопку Send [Отправить].

Пусть для Linux интернет-пейджеры уже имеются в изобилии, многие участники нашей акции Make it With Mono [Сделай это в Mono] (**www.linuxformat.co.uk/makeitwithmono**) просили клонировать Miranda IM. Мы именно этого делать не будем. Мы сделаем нечто похожее на то, что делала Miranda IM в свои первые дни: простой чат-клиент для обмена сообщениями между людьми в вашей сети. Все эти вещи делаются через сокеты, являющиеся объектами сетевого соединения и позволяющие отправлять и принимать сообщения по данному номеру порта.

Сперва два предостережения. Первое: Fedora 7 уже вышла, так что я обновился и рекомендую вам сделать то же – иначе может оказаться, что ваша версия Mono не поддерживает некоторые функции, которые мы будем использовать в этом и будущих уроках. Второе: сокеты – одна из нескольких ужасных частей .NET, и они не просты в изучении. Работая с ними, я всегда пишу небольшую обертку, чтобы остальная часть моей программы не должна была сражаться с сокетами .NET напрямую, но лично я ненавижу, когда кто-то пишет учебник, принуждающий использовать чужие кодовые библиотеки, так что обертку я опущу.

Согласны? Все готовы переодеться хакерами и создавать классные проекты? Тогда вперед!

Создание основы

Годы прослушивания PR от Sun, твердящей, что «сеть – это компьютер», видимо, наконец запали мне в душу – я замечаю, что меня трясет, если мой компьютер отключается от Интернета более, чем на 30 минут. Но хотя на Mono очень легко писать простые приложения для сети, типа нашей RSS-читалки из прошлых номеров (LXF89), написать серьезный проект не так-то просто, потому что для получения достойного результата необходимо использовать асинхронную работу с сетью. Слово «асинхронный» жутко пугает многих, но означает оно всего-навсего то, что оставшаяся часть нашей программы может продолжать себе работать, до тех пор, пока что-то не поступит на сокет. Если это звучит ужасно похоже на многопоточность, то вы попали в точку – именно так и работают асинхронные сокеты.

Проект этого месяца вначале планировалось разбить на три части: посылка и прием данных, создание простого графического интерфейса пользователя, затем создание и того и другого одновременно. Но во время моей работы над объяснением работы кода стало ясно, что для GTK-части не хватит места, и я оставляю ее на вас. Если вы читали наш учебник GTK/RSS, то больших сложностей не должно быть, но в случае проблем просто напишите об этом на нашем форуме **Programming** на **www.linuxformat.co.uk** – и я, или кто-то еще, попытаемся ответить. Итак, приступим к первой и единственной части этого урока...

Запустите MonoDevelop (О-о! Fedora 7 содержит MonoDevelop 0.13 – круто!) и создайте новый консольный проект C# под названием *Chinwag*. Как обычно, убедитесь, что вы настроили использование Runtime version 2.0 в Project > Options > Runtime Options.

В итоге, *Chinwag* будет работать и как клиент (отсылать сообщения другим), и как сервер (принимать сообщения, а затем пересылать их другим клиентам), так что мы разработаем три класса: Chinwag, ChinwagClient и ChingwagServer. Первый необходим просто как основа для двух остальных: при запуске программы ей необходимо проверить слова ‘client’ или ‘server’, чтобы знать, какой режим использовать, и соответствующий класс будет создан и запущен. Итак, перво-наперво создадим класс Chinwag, а также скелет для классов ChinwagClient и ChinwagServer, который мы заполним позднее. Вот код:

```
using System;
using System.Collections.Generic;
using System.Net;
using System.Net.Sockets;
using System.Text;
using System.Threading;
```

Месяц назад Мы скрестили SQL и C#, упростив доступ к базам данных.

чат-клиента

```
namespace Chinwag {
  class Chinwag {
    static void Main(string[] args) {
      if (args.Length == 0) {
        Console.WriteLine("You must specify either 'client' or 'server'.");
        return;
      }

      switch (args[0]) {
        case "client":
          ChinwagClient client = new ChinwagClient();
          client.Run();
          break;
        case "server":
          ChinwagServer server = new ChinwagServer();
          server.Listen();
          server.Run();
          break;
      }
    }
  }

  class ChinwagClient {
    public void Run() { }
  }

  class ChinwagServer {
    public void Listen() { }
    public void Run() { }
  }
}
```

Этот код соберется и запустится, но не будет делать ничего, кроме принуждения вас использовать 'client' или 'server' в качестве параметра командной строки. Он бесполезен, но скоро станет полезным...

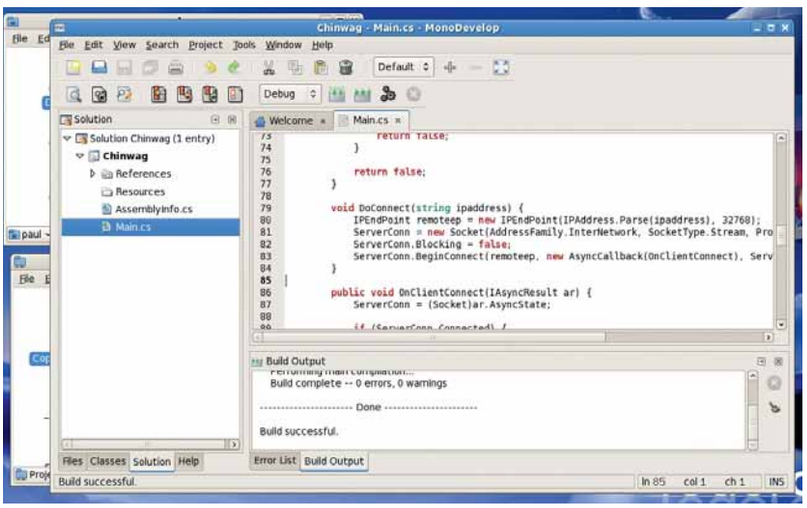
Есть здесь кто-нибудь?

Создавать клиента бессмысленно, пока мы не сделали сервер, так что приступим. Серверу нужно:

1. Слушать клиентов по порту (мы будем использовать порт 32768, его выбрал Майк, любезно поработавший для нас генератором случайных чисел).
2. Принимать соединения клиентов и хранить их в коллекции сокетов.
3. В промежутках, спокойно спать.

Позднее мы добавим возможность отсылать и принимать текст, но пока этого достаточно. Я вставил комментарии для объяснения кода...

```
Socket OurServerConn; // это сокет, который мы будем слушать
List<Socket> RemoteConns = new List<Socket>(); // Это сокеты, соответствующие нашим клиентам
byte[] SocketBuffer = new byte[1024]; // Это пригодится позднее, а пока пропустите!

public void Run() {
  while (true) {
    // Запускается навсегда и ничего не делает
    Thread.Sleep(Timeout.Infinite);
  }
```




```
  }

  public void Listen() {
    // Слушаем порт 32768 для всех IP-адресов
    IPEndPoint local_ep = new IPEndPoint(IPAddress.Any, 32768);

    // Создаем сокет и связываем его с портом
    OurServerConn = new Socket(AddressFamily.InterNetwork, SocketType.Stream, ProtocolType.Tcp);
    OurServerConn.Bind(local_ep);

    // Теперь начинаем слушать, с очередью ожидания до 10 соединений
    OurServerConn.Listen(10);

    // Готовы к принятию новых соединений
    OurServerConn.BeginAccept(new AsyncCallback(OnIncoming Connection), OurServerConn);
  }

  // Это вызывается, как только возникнет соединение
  void OnIncomingConnection(IAsyncResult ar) {
    // Эта ужасная строка захватывает соединение клиента...
    Socket client = ((Socket)ar.AsyncState).EndAccept(ar);
    // ...а затем добавляет его к нашему списку удаленных клиентов
    RemoteConns.Add(client);
    // Готовы к принятию следующих клиентов
    OurServerConn.BeginAccept(new AsyncCallback(OnIncomingConnection), OurServerConn);
    // и печатаем сообщение о статусе
    Console.WriteLine("A new client has connected.");
  }
```

Из этого блока кода вы должны извлечь три основные вещи:

1. Вы говорите BeginЧтото() и сообщаете Mono, какую функцию следует вызвать, когда происходит это событие. Например, для BeginAccept() мы просим Mono вызвать OnIncomingConnection(). »
2. Когда ваш метод отработал, вам необходимо вызвать EndЧтото(). Вы не сможете вызвать BeginЧтото(), пока не вызовете EndЧтото().

› Можно использовать MonoDevelop для серверной части Chinwag, но не клиента; виртуальная оболочка MonoDevelop не умеет читать текст вроде этого.

Скорая помощь

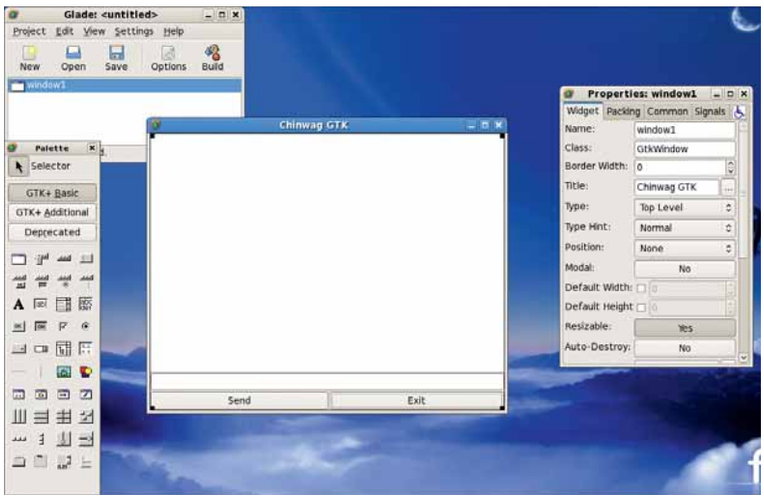
Выполните отладку, используя локальный клиент и локальный сервер, чтобы исключить возможность проблем, связанных с сетью. Только когда вы знаете, что код выполняет любые возможные операции корректно, переходите к отладке в сети.

3 После выполнения функции обратного вызова, вы должны вновь вызвать BeginЧтото().

Вместо «Чтото» можно подставить любой метод асинхронных сокетов в Mono:

```
BeginAccept()/EndAccept(),
BeginConnect()/EndConnect(),
BeginReceive()/EndReceive() и так далее.
```

Вот теперь наш маленький проект начинает работать: если вы откроете терминал, перейдете в каталог **bin/Debug** проекта *Chinwag*, затем выполните *mono Chinwag.exe server*, то программа запустится. Если затем вы откроете окно другого терминала, то сможете попытаться соединиться с сервером, набрав *telnet localhost 32768*. Вы не можете делать что-либо, зато сможете увидеть статусное сообщение 'A new client has connected' [Новый клиент соединен] в окне сервера – по крайней мере, какой-то прогресс налицо!



В этом месяце у нас нет места для создания графического интерфейса на GTK, так почему бы не попробовать вам самим? Лучшие результаты дадут Stetic или Glade (на картинке).

Hello, world

Теперь я хочу заняться клиентом, который имеет нечто общее с сервером. Клиентские соединения производятся с использованием асинхронных методов BeginConnect() и EndConnect(), которые не особо нужны, потому что наш проект не позволяет ничего выполнять, не соединившись. То есть, если мы используем старые добрые синхронные сокеты, для Chinwag разницы не будет; но тогда наш учебник был бы куда менее ценным! Основной цикл для клиента, как и для сервера, это Run(). Он не содержит вызов Sleep(), потому что ему необходимо читать ввод пользователя, который затем отправится по проводам серверу. Это выполняется посредством метода Console.ReadLine() (дополнение к нашему другу Console.WriteLine()) – он читает все, что вводит пользователь до нажатия Enter, обозначающего конец строки.

Вот новый код для *ChinwagClient*, тоже с пояснениями:

```
class ChinwagClient {
  Socket ServerConn;
  byte[] SocketBuffer = new byte[1024]; // Опять игнорируем
  string OurName = "Anonymous"; // Мы собираемся позволить людям изменять их имена позднее

  // Это очень похоже на метод Listen() сервера, кроме того, что вместо BeginConnect() мы используем BeginAccept()
  void DoConnect(string ipaddress) {
    IPEndPoint remoteep = new IPEndPoint(IPAddress.Parse(ipaddress), 32768);
    ServerConn = new Socket(AddressFamily.InterNetwork, SocketType.Stream, ProtocolType.Tcp);
    ServerConn.BeginConnect(remoteep, new AsyncCallback(OnClientConnect), ServerConn);
  }

  // Следующий метод запускается, когда новый клиент успешно подключается к серверу
  public void OnClientConnect(IAsyncResult ar) {
    // Припасем сокет на будущее
    ServerConn = (Socket)ar.AsyncState;
  }

  public void Run() {
    DoConnect("127.0.0.1"); // Соединяемся с локальным сервером
    string input;

    while (true) {
      // Постоянно читаем данные
      input = Console.ReadLine();
    }
  }
}
```

Этот новый код означает, что вы можете теперь запускать различные копии *Chinwag*: одну в режиме сервера и сколь угодно много в режиме клиента. Каждый раз при подключении клиента сервер будет печатать одно и то же сообщение, но мы все еще фактически ничего не делаем. Я же говорил, что сокеты – это непросто, верно?

Главное событие

После колоссальных усилий, мы созрели, чтобы закопаться во внутренности проекта: отправку и получение сообщений между клиентом и сервером. Отправка сообщений производится так:

1 Клиент отправляет сообщения серверу
2 Сервер отсылает сообщения всем другим клиентам
3 Клиенты печатают свои сообщения на экране

С точки зрения клиента, необходимо быть готовым к приему контента, печати любого полученного им текста, а также отправке любого текста, вводимого пользователем. И снова, проще всего показать все это при помощи кода с пояснениями, он идет ниже:

```
// Это старый элемент
public void OnClientConnect(IAsyncResult ar) {
  ServerConn = (Socket)ar.AsyncState;

  // А это новый – вам необходимо вызвать BeginReceive(), если хотите получить текст
  if (ServerConn.Connected) {
    ServerConn.BeginReceive(SocketBuffer, 0, SocketBuffer.Length, SocketFlags.None, OnSocketReceive, ServerConn);
  }
}
```

Скорая помощь

Убедитесь, что вы полностью протестировали ваш код, прежде чем позволить другим его использовать. Под этим я подразумеваю, что вам необходимо задать себе вопросы типа «что если кто-то попытается отослать сообщение в 3 000 букв?». Помните, что сокеты обычно разбивают длинные сообщения, вроде такого, на несколько более коротких, и вам необходимо убедиться, что ваш код объединяет их снова.

Скорая помощь

Системы типа Unix вообще-то позволяют всем пользователям открывать порты с номерами больше 1024 без привилегий root, так что хорошенько подумайте, если хотите использовать порт из этого диапазона. Наш порт специально имеет большой номер – такой уж Майк чудак, но вам нужно позаботиться о выборе порта, еще не занятого другими, как в случае с 6667 или 8080.

Идем дальше

Код на DVD этого месяца содержит небольшое дополнение к нашему проекту для распознавания ввода клиента, который начинается с / в качестве команды. Например, вы вводите /connect 127.0.0.1 для соединения с локальным сервером. Но вы также можете ввести /name Bob для смены вашего имени. Если хотите проверить свои навыки, попытайтесь написать команду /ping, отсылающую короткие сообщения серверу, на которые последний отвечает. Затем можете измерить время отклика и таким образом узнать, хорошо ли работает ваше чат-соединение.

Пересылка двоичных данных

Как показано в этом проекте, наш чат-клиент не может работать с нестроковыми данными, потому что все пересылаемые и получаемые байты преобразуются в строки. Решение этой проблемы – пересылать данные по проводам в виде Base64-шифрованных строк, чтобы данные были в безопасности. Это также позволяет вам смешивать и сравнивать строки и двоичные данные, используя нечто вроде XML, для разделения основной и вспомогательной информации.

```
// Что-то пришло!
void OnSocketReceive(IAsyncResult ar) {
   // Вызываем EndReceive() и перехватываем возвращаемое значение, чтобы увидеть, сколько байтов было послано
   int bytes = ServerConn.EndReceive(ar);
   // Теперь готовимся к приему следующего содержимого
   ServerConn.BeginReceive(SocketBuffer, 0, SocketBuffer.Length, SocketFlags.None, OnSocketReceive, ServerConn);

   // Данные сокета копируются в SocketBuffer (массив байтов, созданных нами ранее) – этот код преобразует байты в строку, пригодную для вывода
   string text = Encoding.ASCII.GetString(SocketBuffer, 0, bytes);
     Console.WriteLine(text);
}

// Пересылка текста по проводам противоположна его приему: нам необходимо преобразовать введенную строку в байты
 public void SendText(string text) {
   // Это шикарный способ сборки строки из переменных
   text = string.Format("<{0}> {1}", OurName, text);
   ServerConn.Send(Encoding.ASCII.GetBytes(text));
}
```

Это дает вам основу для пересылки текста от клиента серверу, но чтобы действительно заставить текст идти от пользователя к серверу, нужно модифицировать клиентский цикл Run() так:

```
while (Running) {
   input = Console.ReadLine();
   SendText(input);
}
```

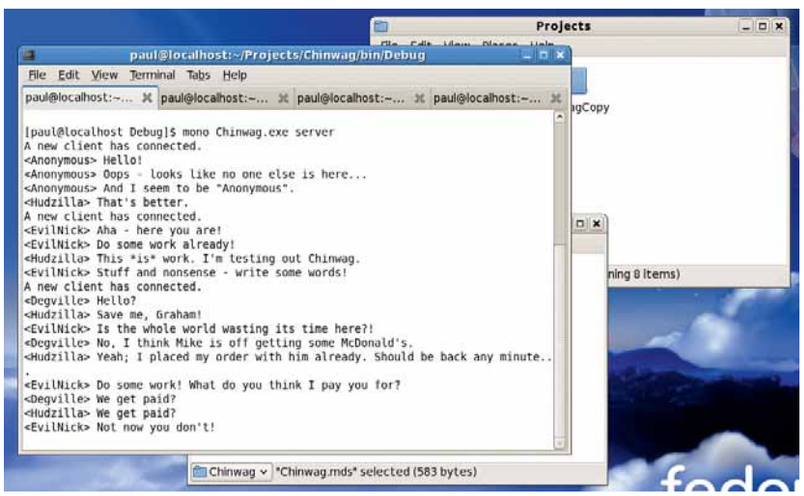
Теперь осталось сделать только одну вещь: обновить сервер так, чтобы он мог получить текст и также отправить полученный текст другому клиенту. Это очень похоже на код отправки и получения текста у клиента, так что объяснять тут особо нечего:

```
void OnIncomingConnection(IAsyncResult ar) {
   Socket client = ((Socket)ar.AsyncState).EndAccept(ar);
   RemoteConns.Add(client);
   // Это немного новое – нам необходимо быть готовым к приему данных от клиентов
   client.BeginReceive(SocketBuffer, 0, SocketBuffer.Length, SocketFlags.None, OnSocketReceive, client);
   OurServerConn.BeginAccept(new AsyncCallback(OnIncoming Connection), OurServerConn);

   Console.WriteLine("A new client has connected.");
}

// Клиент отослал текст!
void OnSocketReceive(IAsyncResult ar) {
   Socket client = (Socket)ar.AsyncState;
   // Это то же самое, что и в коде клиента
   int bytes = client.EndReceive(ar);
   client.BeginReceive(SocketBuffer, 0, SocketBuffer.Length, SocketFlags.None, OnSocketReceive, client);

   string text = Encoding.ASCII.GetString(SocketBuffer, 0, bytes);
   Console.WriteLine(text);
   // за исключением этой части: нам необходимо отправить текст всем другим клиентам
   SendToAll(client, text);
}

void SendToAll(Socket from, string text) {
   // Нам необходимо отправить текст каждому клиентскому сокету, КРОМЕ того, кто изначально его отправил
   foreach (Socket sock in RemoteConns) {
     if (sock == from) continue;

     sock.Send(Encoding.ASCII.GetBytes(text));
   }
}
```

Вот и все. Это очень сложный проект, поскольку он работает с клиентами и сервером в одном файле исходного кода. Работа с каждым из них по отдельности была бы значительно легче, но зато теперь у вас есть инструменты, необходимые для самостоятельного создания любых программ. Если вы сможете создать нечто крутое и интересное – вроде мини web-сервера, пересылки файлов или, возможно, просто улучшенной версии этой чат-системы –скиньте мне ваш исходный код, лицензированный по свободной лицензии (сгодятся GPL или BSD), и я посмотрю, можно ли его выложить на один из наших дисков-приложений к Linux Format. Это если я не увлекусь захватывающим сетевым чатом... LXF

Скорая помощь

Telnet – прекрасный способ протестировать этот проект, особенно если у вас есть ошибки, и вы не уверены, в чем проблема: в клиенте или в сервере. Однако стоит вам усложнить протокол (например, перейти на XML), как вы обнаружите, что польза от Telnet быстро снижается.



› Несколько клиентов могут подключиться к одному серверу и обмениваться сообщениями в командной строке. В следующем месяце я, возможно, поработаю над черным списком, чтобы спрятать ваш чат от босса...

Через месяц Шифрование на уровне правительства – это просто. Нет, правда.

APT: Хитрости

Как раз когда вы подумали, что можете без опаски установить ПО, **Пол Хадсон** решил показать вам несколько мощных возможностей APT...

Наш эксперт

Пол Хадсон
Кто же лучше напишет об APT, чем парень, работающий над ним? К сожалению, мы не смогли заставить его сделать это для нас, и Пол любезно вызвался добровольцем.

Первая буква в APT означает “advanced” [продвинутый], но если вы используете для него легковесную оболочку типа Synaptic, то он вдруг оказывается простоватым на вид, как будто он ничем не лучше RPM. Но вы ведь неспроста используете дистрибутив на базе Debian, верно? Конечно, если причина в любви к оформлению, с которым по умолчанию поставляется Ubuntu, то, вероятно, вам лучше перейти к отличному учебнику для начинающих на стр. 56.

Итак, похоже, остались одни профессионалы. Позвольте мне обрисовать то, что вы собираетесь терзать и брать на заметку в этом месяце. Вы изучите...

- Как создать хранилище APT и восстановить настройки ваших приложений
- Как автоматически выбрать самое быстрое зеркало
- Как очистить архив пакетов для экономии места на диске
- Как автоматически заполучить и собрать пакеты исходных текстов
- Как использовать *apt-cache* и *apt-file* для поиска пакетов
- и более!

Если ваш APT-опыт заключается лишь в установке программ, то вы узнаете многое. Если вы никогда до этого не использовали APT (без удобной оболочки), то вы узнаете еще больше. Сидите у моих ног и слушайте, пока я объясняю…

Самые основы

Advanced Packaging Tool [Продвинутый пакетный инструментарий] – это просто библиотека, известная как *Libapt*. Она также является оболочкой, поскольку всю тяжелую работу берет на себя *dpkg* – настоящее ядро менеджера пакетов Debian. APT отслеживает зависимости пакетов, вызывает для вас *dpkg* и запросто устанавливает приложения.

Итак, простая строка «установи мне программу» выглядит так:

```
apt-get install foobar
```

Если вы не можете найти пакет *foobar*, поищите командой

```
apt-cache search foobar
```

…или, в менее строгой форме, просто

```
apt-cache search foo
```

Проблема с нестрогостью, однако, в том, что вы получите всевозможные дурацкие совпадения – вышеприведенному запросу соответствует такая вещь, как *lighttpt*, потому что в его описании сказано «minimal memory footprint». Конечно, искатели поумнее используют перед шаблоном символ «крышки», например, так:

```
apt-cache search ^foo
```

…означающий, что искомое слово должно содержаться в начале имени или описания пакета. Аналогично, указание $ в конце говорит, о том, что искомое слово должно быть в конце имени или описания пакета.

Теперь, используя ваши невероятные навыки APT, запустите от имени суперпользователя (или используйте *sudo* в Ubuntu):

```
apt-get install apt-file
```

Именно в этот момент вы чудесным образом превращаетесь из героя командной строки в супергероя: *apt-file* позволяет вам запрашивать, к какому пакету принадлежат файлы или опрашивать пакет о содержащихся в нем файлах. Перед тем как начать, выполните *apt-file update*, потому что так собирается информационный кэш файлов. Теперь попытайтесь выполнить следующее и посмотрите, что происходит:

```
apt-file search apache2ctl
```

Будут просмотрены все доступные в вашей системе пакеты в поисках того, в котором находится файл **apache2ctl**, и в результирующем списке вы получите **apache2.2-common**, потому что он содержит этот файл. Вы можете также выполнять обратный поиск, то есть попросить *apt-file* вывести все файлы, содержащиеся в данном пакете:

```
apt-file list apache2.2-common
```

Upgrade и dist-upgrade

В APT есть два способа обновления программ: apt-get upgrade и apt-get dist-upgrade. Первый ищет обновления для всех программ в вашей системе, но не выполняет обновление, если для этого требуется установить или удалить другие пакеты для удовлетворения зависимостей. Второй обновляет все программы, даже если это требует каких-то добавлений или удалений, и это делает его фактически необходимым для обновления с одной версии дистрибутива до другой. APT относительно умен в обновлении приложений, потому что, делая выбор между обновлениями, выставляет приоритеты для наиболее важных программ.

Месяц назад Мы писали скрипты для менеджера виртуальных рабочих столов в KDE.

умельцев

Итак, когда в следующий раз вы увидите подозрительный файл где-то в **/bin** или **/usr/bin**, обратитесь к своему приятелю *apt-file*.

Сила зеркал

Если вы часто устанавливаете приложения, можно сберечь многие часы вашего времени, перенастроив APT на выбор зеркала с большей скоростью загрузки. А каждый сэкономленный час – это час для игры в *Crack Attack*, так что я уверен: быстрейшее зеркало вам обязательно нужно.

Если у вас чистый Debian, то в нем есть инструмент для зеркал, который выполнит за вас всю тяжелую работу: *netselect-apt*. Он основан на другом инструменте, называемом просто *netselect*, который прозванивает список серверов, проверяя время отклика. *netselect-apt* идет дальше, а именно а) загружает список серверов с debian.org, б) прозванивает их все, в) выбирает самый быстрый сервер и г) автоматически записывает новый файл sources.list для использования этого последнего.

Итак, выполните *apt-get install netselect-apt* для установки приложения, затем просто запустите *netselect-apt* и дайте ему поколдовать несколько минут. По окончании он запишет новый файл **sources.list** в текущий каталог, и вам надо открыть его и текущий файл **sources.list** один рядом с другим, чтобы вы могли вставить новый файл в существующую конфигурацию.

Если у вас не Debian, то, вероятно, у вас Ubuntu, а там имеется собственная прекрасная версия *netselect-apt*, встроенная в *Synaptic*. Поэтому:

1. Запустите *Synaptic* из Система > Администрирование > Менеджер пакетов Synaptic.
2. Перейдите в строке меню Настройки > Репозитории
3. На вкладке Программное обеспечение Ubuntu, в списке выбора под заголовком Загрузить с выберите Другой...

В этом окне можно выбрать из списка конкретное зеркало в зависимости от страны, где вы находитесь, или щелкнуть на кнопке Выбрать лучший сервер, чтобы Ubuntu запустил *netselect* на основе списка серверов Ubuntu. Это потребует несколько секунд, но зато вы гарантированно получите наибыстрейший сервер из вам доступных.

Простое переформатирование

Что вам больше всего нравится в том, что менеджер пакетов управляет вашими программами? Бесспорно, две чаще всего указываемые функции – это легкость установки новых программ и автоматическое обновление, но одна из моих любимых – это возможность сохранить список выбранных вами программ в файле, чтобы прочитать позднее. На самом деле, будучи крупным потребителем виртуальных машин на нескольких ПК и имея возможность создавать список стандартных программ, которые мне нужно установить, я не должен в самый неподходящий момент проверять, не забыл ли какую-нибудь программу, потому что каждый ПК имеет одинаковый набор установленного ПО!

В APT эта магическая суперсила называется *dpkg --get-selections*. Он просматривает все установленные у вас программы и выводит их на экран. Например:

```
dpkg --get-selections > software
```

Здесь список установленных программ будет сохранен в файле **software**: все до одного приложения, необходимые для воспроизведения состояния вашего ПК, теперь в нем. И если у вас есть резервная копия ваших документов (например, вашего каталога **/home**), то в комбинации с данным списком – это все, что необходимо для полного восстановления. Да, я понимаю, что многие пользователи Windows думают, что для безопасности им необходимо резервировать весь диск **C:**, но у нас в Linux нет такой проблемы: в **/bin**, **/sbin**, **/usr** и им подобные все ПО поступает из менеджера пакетов – вот пусть он и помнит, какие файлы нам нужны!

› Графическая версия netselect-apt в Ubuntu встроена в Synaptic – просто нажмите кнопку Select Best Server [Выбрать лучший сервер] и подождите, пока он выполнит сотни пингов от вашего имени.

Итак, сохранять список программ мы умеем, но ведь надо и восстанавливать этот список! Для этого вам необходим брат-близнец *--get-selections*, а именно (вы знали, вы знали!) *--set-selections*. Он читает данные пакетов строку за строкой, и мы закачаем в него содержимое **pkglist** – вот так:

```
dpkg --set-selections < pkglist
```

Ваш раздел приложений будет настроен, но на самом деле ничего не установится. Для этого необходимо выполнить специальную команду *apt-get*, которая означает «посмотри список программ, которые следует установить, и сравни его с тем, что уже установлено, затем выполни необходимые изменения»:

```
apt-get dselect-upgrade
```

Любое приложение, имеющееся в списке выбора, но не установленное, будет теперь установлено, так что идите попейте кофе и возвращайтесь обратно – у нас еще много магической работы!

Очистка пространства

Типы вроде меня устанавливают много приложений. Мой список *--get-selections* содержит более 3000 пакетов, так что после установки они занимают много места на диске. Но «для вашего удобства» APT любезно хранит все сжатые пакеты для всех установленных программ, чтобы очень быстро переустановить, если вы их удалили. На это уходит много места, потому что каждая программа съедает в два раза больше дискового пространства, чем ей нужно!

Посмотрите на каталог **/var/cache/apt/archives**. Если вы установили ваш дистрибутив сравнительно давно, то, вероятно, этот каталог содержит несколько сотен мегабайт ПО, которое бесполезно, если только вы не переустанавливаете программы на каждом шагу, чего большинство »

» из нас не делает. Можно бы просто выполнить *rm **, но это плохое решение: частично загруженные пакеты удалены не будут. Поэтому APT предоставляет нам два пути удаления этих старых пакетов:

» *apt-get autoclean*

» *apt-get clean*

Первый автоматически удалит архивные пакеты, которые сочтет старыми (то есть вы устанавливали их так давно, что шансы на то, что они вам нужны, невелики). Второй просто удаляет все архивные пакеты, старые или нет, освобождая, по существу, все занимаемое ими дисковое пространство. Если вам по жизни надо дурковать с недавно установленными пакетами, то опция *autoclean* как раз для вас. В противном случае обойдитесь простым *clean*.

Уборка мусора

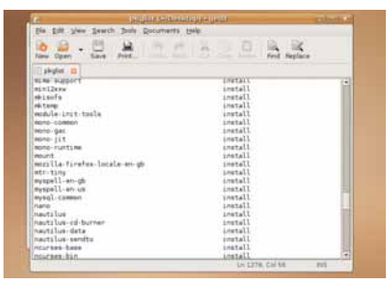
Приложения удаляются командой *apt-get remove имяпакета*, но на самом деле это только половина работы: приложение-то удаляется, но остаются файлы его конфигурации. На вид это неумно, тем более что одна из самых ненавистных вещей в Windows – манера оставлять файлы, разбросанные повсюду в вашей системе! Но на практике это означает, что если вы когда-либо переустановите приложение, оно подключит и будет использовать существующие файлы конфигурации, так, как будто вы его не удаляли.

Если это раздражает вас, или вы помешаны на экономии каждого байта вашего дискового пространства, то вам необходима опция *--purge* для *apt-get remove*, заставляющая APT удалить все, что установила программа. Если вы опытный пользователь *Synaptic*, то *--purge* соответствует «Пометить для полного удаления».

Другая область, где APT может облегчить жизнь вашей системе – установка зависимостей ПО. Они явно необходимы, потому что без зависимостей ваши приложения не будут работать. Но что произойдет, если вы удалите некую программу? Я скажу вам, что: все зависимости, шедшие с ней, останутся в вашей системе, даже если они больше не нужны. Подумайте, сколько места на диске тратится даром!

И вновь APT спешит на помощь, со специальной командой: *apt-get autoremove*. Она сканирует вашу базу данных пакетов на предмет зависимых пакетов, которые больше не нужны, и удаляет их. Это сбережет немало пространства, особенно если вы устанавливали все библиотеки KDE для запуска уже неактуальных KDE-программ; но, с другой стороны, когда все эти зависимости понадобятся, их придется загружать снова. Помните, место на диске дешево.

› Выполнение *dpkg --get-selections* – наискорейший способ доводки новой системы.



Наверное, большинство людей использует APT ради его двоичных пакетов – тех классных предварительно собранных программ, которые «просто работают», потому что скомпилированы в системе, похожей на вашу. Но если вы иногда тоскуете по собирабельности (вот вам кандидат в орфографический словарь) Gentoo, то у Debian есть интересный компромисс: он позволяет вам, при желании, собирать пакеты самостоятельно. Нет, это не трудно – и да, вы получите все преимущества от сборки программ из исходных текстов. Так и вправду лучше, поскольку вы получите все дополнительные модификации, предусмотренные Debian для ПО, и оно действительно будет соответствовать двоичной версии, которую вы можете установить.

› Вот что вы видите при запуске dselect. Будьте осторожны – очень осторожны! Зато это не так плохо, как Aptitude...

Прямиком к истокам

Для начала позаботьтесь о настройке вашего APT на загрузку файлов с исходными текстами из репозитория. Если вы используете чистый Debian, это означает, что вам необходима строка deb-src для каждой deb-строки в вашем файле **sources.list**. Если вы в Ubuntu, то вернитесь обратно в окно репозиториев в *Synaptic* и убедитесь, что выбран пункт Исходные тексты.

Теперь вы можете загрузить исходный текст пакета так:

```
apt-get -b source anjuta
```

Выполнив эту команду – или ей подобную, для любого желаемого пакета – вы, вероятно, получите сообщение об ошибке типа «E: Child process failed» [Ошибка дочернего процесса]. Это практически бессмысленно, но если вы немного прокрутите вверх, то увидите нечто вроде 'dpkg-checkbuilddeps: Unmet build dependencies', означающее, что у вас установлено не все необходимое ПО, чтобы можно было собрать *Anjuta*.

Если бы вы компилировали *Anjuta* вручную, вам бы пришлось просмотреть список зависимостей и установить их одну за другой. Но с APT есть способ намного легче: *apt-get build-dep*. Он позволяет вам указать имя пакета и загрузить все требуемое ПО, чтобы сделать возможным сборку пакета из исходных текстов. Например:

```
apt-get build-dep anjuta
```

Эта команда попросит APT загрузить требуемые библиотеки и инструменты и соберет все необходимое для компиляции ПО *Anjuta*, и вы сможете выполнить *apt-get -b source anjuta* вновь: на этот раз оно будет собрано из исходных текстов.

По завершении (а времени может уйти немало, в зависимости от размера компилируемого ПО), *apt-get* создаст пакет **.deb**, который можно установить напрямую, командой:

```
dpkg -i anjuta-some-version.deb
```

Он установится при помощи менеджера пакетов, как и обычное приложение, но будет на 100% оптимизирован именно для конфигурации вашей системы.

Для вещей вроде *Anjuta* это в принципе не важно, но для приложений, зависящих от процессора, типа *FFmpeg* или *Mplayer*, отдача может быть намного большей – попробуйте сами! LXF

Скорая помощь

Вам нужна сила супер-коровы? Достаточно сказать `apt-get moo`

Истинно продвинутые интерфейсы

Некоторые думают, что *dselect* – это хороший способ взаимодействовать с APT. Однако некоторые также думают, что версия Mr Tambourine Man от Уильяма Шатнера [William Shatner] тоже хорошая, поэтому я думаю, что нельзя угодить всем. Но если вам нравится мучительный, кошмарный опыт, когда ярлыки на экране кажутся не соответствующими тому, что вы на самом деле хотите сделать, то запустите *dselect* от имени суперпользователя. Это классическая оболочка командной строки для APT, и, в принципе, это *Synaptic* для командной строки. Новая, «улучшенная» оболочка – это *Aptitude* (просто выполните aptitude в командной строке), но, по нашему мнению, она еще непонятнее, чем *dselect*.

» Через месяц Научитесь перекомпилировать Gentoo с флагом -f-upgrade-skills.

Мощность.
Надежность.
Производительность.

ETegro technologies

СЕРВЕРА. СИСТЕМЫ ХРАНЕНИЯ ДАННЫХ.
ГРАФИЧЕСКИЕ СТАНЦИИ

Компания ETegro Technologies — производитель системного программного обеспечения и аппаратных решений: серверов, графических станций, кластеров, систем хранения данных.

В компании работают высококлассные специалисты, имеющие опыт работы в ведущих IT-компаниях России и других стран.

ETegro Technologies занимается разработкой программного и аппаратного обеспечения по заказу государственных и частных компаний.

Более подробную информацию о компании ETegro Technologies можно получить на сайте www.etegro.com.

Гарантии сохранности ваших данных.

Кстати.

Сравните цены с аналогами.

Центральный офис
Москва, Электродная ул., д. 2, стр. 12-13-14
Телефон: +7 (495) 380-02-88
Факс: +7 (495) 380-02-88

E-mail: sales@etegro.com
www.etegro.com

Реклама

Перекличка серверов

ЧАСТЬ 8 Отыскали пару лишних серверов и готовы написать настоящее распределенное приложение? **Александр Бабаев** подскажет подходящий способ.

На этом уроке наше приложение будет усложнено ровно в два раза. Раньше сервер был один (он обрабатывал запросы пользователей и менял список записей телефонной книги), теперь добавим ему напарника. Предположим, что это выделенный сервер, который хранит данные пользователей – имена, пароли, дополнительную информацию, и мы решили связать web-сервер с этим вторым, чтобы, например, можно было по имени пользователя получить дополнительную информацию о нем.

Для удобства ссылок дадим серверам названия. Пускай наш старый называется Нео, а большой сервер с разной информацией – Матрица.

Прикинем, как все устроить. Для тестов все программы (оба сервера) будут запускаться на одном компьютере. Но, поменяв соответствующие IP-адреса (ниже мы разберемся, какие именно), можно будет запускать Нео и Матрицу на разных компьютерах одной и той же подсети. Правда, если между ними есть всякие-разные устройства вроде маршрутизаторов, шлюзов и мостов, тесты могут и не заработать. Это предупреждение! Реализовать более сложную схему взаимодействия тоже можно, но на это нам, к сожалению, не хватит журнального места, которое вполне конечно.

Итак, получается следующая топология сети:

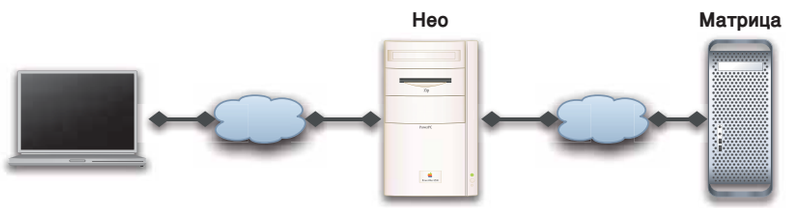


Рис. 1. Топология сети.

Как докричаться?

Классический способ связи двух разных программ – через файлы: одна пишет, другая читает. Например, можно сделать общий файл для записи команд, куда пишет Нео и откуда читает Матрица, и файл ответов, где запись/чтение происходят наоборот.

Способ простой, но плохой: возникает огромное количество проблем, из которых падение производительности – самая маленькая. Рассматривать его не будем.

Другой способ – связь через так называемый «сокет». Сокет – это такая штука, к которой можно присоединиться [англ. socket – «розетка»]. Сервер «открывает» сокет и «слушает» его. Клиент – подключается к серверному сокету, передает/получает данные, отключается. Затем подключается следующий клиент.

Можно представить себе сокет как телефон. Вы получаете номер абонента (открываете сокет), подключаете телефон (теперь вы слушаете сокет), вам звонят – соединяются с вашим сокетом… Ну, вроде бы понятно? Примерно так же соединяются и компьютеры. А чтобы определить, к какому конкретно сокету присоединяться, нужно знать адрес IP-компьютера и номер порта, на котором открыт сокет. Номер порта необходим, потому что на одном и том же компьютере может быть запущено несколько «слушающих» программ (серверов), и нужно, чтобы они не мешали друг другу. А клиент, с другой стороны, должен точно указать, какой именно сервер ему нужен.

Есть еще и третий способ, «каналы», но о нем мы поговорим несколько позже.

Сокеты

Итак, общая схема работы с сокетами достаточно проста. Сервер создает сокет:

```
ServerSocket serverSocket = new ServerSocket();
```

Дальше нужно открыть сокет для приема соединений. В Java это делается так:

```
serverSocket.bind(new InetSocketAddress("localhost", 10000));
```

Здесь мы говорим, что сервер будет слушать соединения, в адресе назначения которых указано "localhost" и порт 10000. Сразу отметим: чтобы создать сервер на портах с номерами, меньшими 1024, обычно нужны права суперпользователя.

Далее нужно ловить подключения. Обычно это делается в цикле (получили подключение, обработали, ждем следующее). Например, так:

```
while (true) {
    Socket socket = serverSocket.accept();
    int byteReceived = socket.getInputStream().read();
    OutputStream outputStream = socket.getOutputStream();
    outputStream.write(byteReceived + 1);
    socket.close();
}
```

serverSocket.accept() ждет подключения, блокируя текущий поток, и как только оно появится, создает так называемый клиентский сокет, продолжая выполнение потока. Я читаю из потока сокета один байт, записываю в сокет его же плюс единицу и закрываю сокет.

Вот так все несложно. А как устроен клиент? Он создает сокет, пишет в него единицу, читает ответ (двойку) и выводит ее на экран. По Jav'овски это выглядит так:

```
Socket socket = new Socket("localhost", 10000);
socket.getOutputStream().write(1);
int result = socket.getInputStream().read();
System.out.println("Result is: " + result);
socket.close();
```

Здесь тоже все предельно просто, но тем не менее, работает. То есть, если создать два класса и в main первого поставить код сервера, запустить (он будет «висеть», слушая соединения), в main второго – код клиента, и запустить… Клиент выдаст 2.

NIO

Проблемы начинаются, когда соединений больше одного, когда их одновременно больше одного, когда они достаточно длительные, когда нужно принимать/отсылать громадные файлы или протокол работы

Месяц назад AJAX в ритме Java – мы прошли экспресс-курс по написанию приложений Web 2.0.

сложный. Если все делать вручную, как мы сейчас, то получается очень и очень трудоемко: огромное количество потоков (рекомендуется, собственно, по одному на каждый запрос), проблемы отсоединений клиентов, отслеживания этих событий, сериализация (запись в поток и обратно) объектов, и так далее.

Еще один минус блокирующих сокетов в том, что они (простите за тафтологию) блокируют поток в ожидании подключения. А если подключения нет? Значит, поток заблокирован навсегда. И дело не только в подключениях, но и в записи/чтении в/из сокета: случись что во время этих операций – заблокируют навсегда. Вдобавок при большом количестве соединений/потоков скорость работы сервера уменьшается гораздо быстрее, чем увеличивается количество соединений. То есть имеет место типичная ситуация плохой масштабируемости.

И тут на помощь приходит новый способ обработки соединений – NIO (The new I/O API), хотя и более сложный, но гораздо более масштабируемый. Эта библиотека появилась в Java 1.4, и она позволяет создавать неблокирующие соединения.

Рассмотрим тот же сервер и клиент в реализации NIO.. Сначала создадим так называемый «канал»:

```
Selector acceptSelector = SelectorProvider.provider().openSelector();
ServerSocketChannel socketChannel = ServerSocketChannel.open();
socketChannel.configureBlocking(false);
```

Мы не просто создали канал, но еще и объявили его неблокирующим (работа с блокирующими каналами ничуть не лучше работы с блокирующими сокетами). Теперь нужно «объяснить» каналу, какие события нам интересны (присоединение клиента, чтение, запись, отсоединение).

Займемся подсоединением клиента:

```
socketChannel.socket().bind(new InetSocketAddress("localhost", 10000));
socketChannel.register(acceptSelector, SelectionKey.OP_ACCEPT);
```

Теперь начинаем волшебный цикл обработки. В цикле проверяем, есть ли интересующие нас события в «канале», и если есть, итерируем по ним, получая клиентский сокет. Дальше с сокетом работаем так же, как было описано выше.

```
while (acceptSelector.select() > 0) {
    Set readyKeys = acceptSelector.selectedKeys();
    Iterator i = readyKeys.iterator();

    while (i.hasNext()) {
        SelectionKey key = (SelectionKey) i.next();
        i.remove();
        ServerSocketChannel nextReady = (ServerSocketChannel) key.channel();
        Socket socket = nextReady.accept().socket();
        outputInputPlusOne(socket);
    }
}
```

Функция outputInputPlusOne(socket), как мы делали ранее, читает число, увеличивает его на 1 и записывает результат.

При этом поток постоянно крутится в цикле, не блокируясь. Это позволяет использовать достаточно интересные методы обработки подключений: например, создавать только один поток для произвольного количества подключений, и он один будет их все обрабатывать. Плюс, операции чтения и записи тоже можно сделать неблокирующими, и совсем будет сказка… Если бы не сложность. Переход на неблокирующую обработку подключений – увеличение объема программы в три раза. Неблокирующая запись и чтение – еще больше. В результате получается достаточно громоздкая схема. Увы, только так можно добиться хорошей производительности.

Минное поле

Но есть замечательная библиотека, где все это уже учтено. Внутри она использует NIO, поэтому отлично масштабируется. Зато наружу выходят очень простые и понятные методы, которые можно использовать, получая превосходный результат. Библиотека называется *Apache Mina*, она достаточно широко распространена (например, Jetty, который мы рассматривали в первой статье цикла, использует именно ее) и неплохо отлажена. Иными словами, ею можно пользоваться в «промышленных» масштабах, не особо беспокоясь о проблемах.

Сервер делается так:

```
SocketAcceptorConfig acceptorConfig = new SocketAcceptorConfig();
acceptorConfig.getFilterChain().addLast("Serializator",
        new ProtocolCodecFilter(new ObjectSerializationCodecFactory()));
DataServerHandler handler = new DataServerHandler();
SocketAcceptor acceptor = new SocketAcceptor();
acceptor.bind(new InetSocketAddress("localhost", 10000), handler, acceptorConfig);
```

В первой строчке создается конфигуратор (через него задаются разные параметры подключения), затем в него вставляется фильтр, через который пройдут все данные (фильтр распаковывает объекты Java из потока). В третьей строке создается обработчик событий (о нем чуть ниже), который обрабатывает события *Mina*, и дальше – инициализируется сервер (опять тот же bind).

Поскольку *Mina* использует NIO, сервер получается событийный. При подключении клиента вызывается обработчик (Handler), который обслуживает запрос. При поступлении данных – тоже вызывается Handler (уже другой метод). И так далее. Вот как выглядит обработчик (Handler) в нашем случае:

```
public class DataServerHandler extends IoHandlerAdapter {
    public void messageReceived(IoSession session, Object message) throws Exception {
        if (message instanceof Integer) {
            session.write(((Integer) message) + 1);
            session.close();
        }
    }
}
```

Мы переопределили только один метод, который отвечает за получение данных. *Mina* работает не с потоками, а с объектами (преобразованием занимается фильтр, который мы вставляли в конфигуратор). Поэтому наша задача упрощается: читаем Integer, прибавляем единицу, записываем обратно, закрываем сессию. Все.

Предыдущий клиент, правда, не подойдет – нужно создавать другой. Но он не сложнее, чем был. Вот его код:

```
SocketConnector connector = new SocketConnector();
SocketConnectorConfig config = new SocketConnectorConfig();
config.getFilterChain().addLast("Serializator", new ProtocolCodecFilter(new ObjectSerializationCodecFactory()));
ConnectFuture connectFuture = connector.connect(new InetSocketAddress("localhost", 10000),  new DataClientHandler(), config);
connectFuture.join();
IoSession session = connectFuture.getSession();
session.write(new Integer(1));
```

Код клиента – практически точное повторение кода сервера. Даже обработчик есть. Только вместо Acceptor'а создаем Connector. connectFuture – это объект, который позволяет, во-первых, дождаться, пока присоединимся (.join()), а во-вторых, от него получается сессия, куда можно писать всякие объекты (в нашем случае – Integer). Обработчик, как и на сервере, простой:

```
public class DataClientHandler extends IoHandlerAdapter {
    public void messageReceived(IoSession session, Object message) throws Exception {
        System.out.println("Result is: " + message.toString());
    }
}
```

Все. Сложность в данном конкретном случае – четыре класса вместо двух. Но это оправданно. *Mina* позволяет гибко настраивать обработку данных, бегающих между клиентом и сервером: количество потоков, обрабатывающих каналы, и так далее. Все это позволяет создавать простые серверы буквально за минуты, а сложные – за считанные часы или за несколько дней.

RMI

Еще один способ соединения двух Java-программ – RMI (Remote Method Invocation, удаленный вызов метода). Это несколько другой класс соединений, который скрывает сокеты и прочую внутреннюю механику от программиста, но требует понимания совершенно других принципов.

Для работы RMI использует так называемый *registry*. Это специальная служебная программа, которая аккуратно регистрирует классы, желающие быть серверами. Например, там может зарегистрироваться наш сервер. После чего клиент, зная, где находится *registry*, подключается к нему и просит выдать ссылку на сервер – и получает ссылку на класс.

После этих магических пассов методы сервера можно вызывать, как будто они находятся прямо в клиенте, в той же Java-машине. А RMI сам понимает, что вызван метод сервера, организует передачу/прием данных, преобразования всего из всего и выдает результат. На **Рис. 2** приведена схема работы RMI (исключая *registry*):

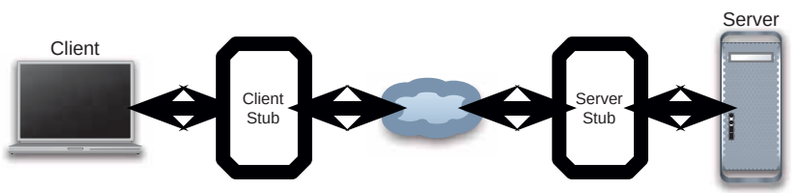


Рис. 2. Схема работы RMI.

При создании класса, который может быть вызван удаленно (схема, когда обращаются непосредственно к методу, а он скрытыми путями вызывается на удаленной машине, называется удаленным вызовом, Remote Procedure Call), генерируются специальные классы-заглушки (stubs). Они-то и делают все «взмахи волшебными палочками». На самом деле, когда клиент выполняет удаленный вызов, вызывается аналогичный серверному метод заглушки. В этом методе параметры сериализуются и через сокет передаются на сервер, где вновь десериализуются, вызывается сервер, а ответ передается таким же образом (через заглушки) клиенту.

Чтобы RMI заработал, нужно сделать интерфейс для сервера и его реализацию, написать клиент, потом сгенерировать заглушки, все скомпилировать и запустить.

Интерфейс сервера прост:

```
public interface IRMIServer extends Remote {
    public abstract int increment(int aValue) throws RemoteException;
}
```

Он должен наследоваться от интерфейса Remote, и все методы, вызываемые удаленно, должны выбрасывать исключение RemoteException.

Реализация сервера не менее проста:

```
public class RMIServer implements IRMIServer {
    public RMIServer() throws RemoteException {
        super();
    }

    public int increment(int aValue) throws RemoteException {
        return aValue + 1;
    }
}
```

Реализация должна обязательно иметь конструктор по умолчанию, который выкидывает исключение RemoteException и вызывает super(). Метод реализуется как обычно.

Регистрация сервера может быть выполнена, например, в методе main того же сервера:

```
public static void main(String[] args) throws Exception {
    System.setSecurityManager(new RMISecurityManager());
    RMIServer Server = new RMIServer();
    Naming.rebind("RMIServer" , Server);
}
```

Здесь устанавливается менеджер безопасности (иначе нам скажут, что нет прав для работы с RMI), создается сервер и регистрируется в *registry* (rebind). Есть и метод bind, но он выдаст ошибку, если такой сервер уже зарегистрирован. В целях экономии места используем rebind, который проверку на существование такого же сервера не выполняет.

Клиент тоже устанавливает менеджер безопасности, после чего в registry ищет сервер (имя он знает). Как только найдет – просто вызывает метод, как если бы RMI не существовало.

```
public class RMIClient {
    public static void main(String[] args) throws MalformedURLException,
NotBoundException, RemoteException {
        System.setSecurityManager(new RMISecurityManager());
        String url = "//localhost/RMIServer";
        IRMIServer remoteObject = (IRMIServer) Naming.lookup(url);
        System.out.println("Result is " + remoteObject.increment(1));
    }
}
```

Запуск сервера и клиента несколько более сложен, чем обычно. До сих пор мы компилировали и запускали классы «просто так». Теперь создадим заглушки и запустим *registry*. Это делается так:

```
rmic RMIServer
```

создаст заглушку сервера (сначала нужно все скомпилировать, а потом выполнить эту команду с полным именем класса сервера). А команда

```
rmiregistry &
```

запустит RMI Registry с параметрами по умолчанию. Чтобы разрешить менеджеру безопасности делать все что угодно, нужно запустить клиент и сервер с параметром, указывающим на файл политики безопасности. Назовем этот файл **open.policy** и запишем в него следующее:

```
grant {
    permission java.security.AllPermission;
};
```

Теперь запустим клиент и сервер:

```
java -Djava.security.policy=open.policy RMIServer
```

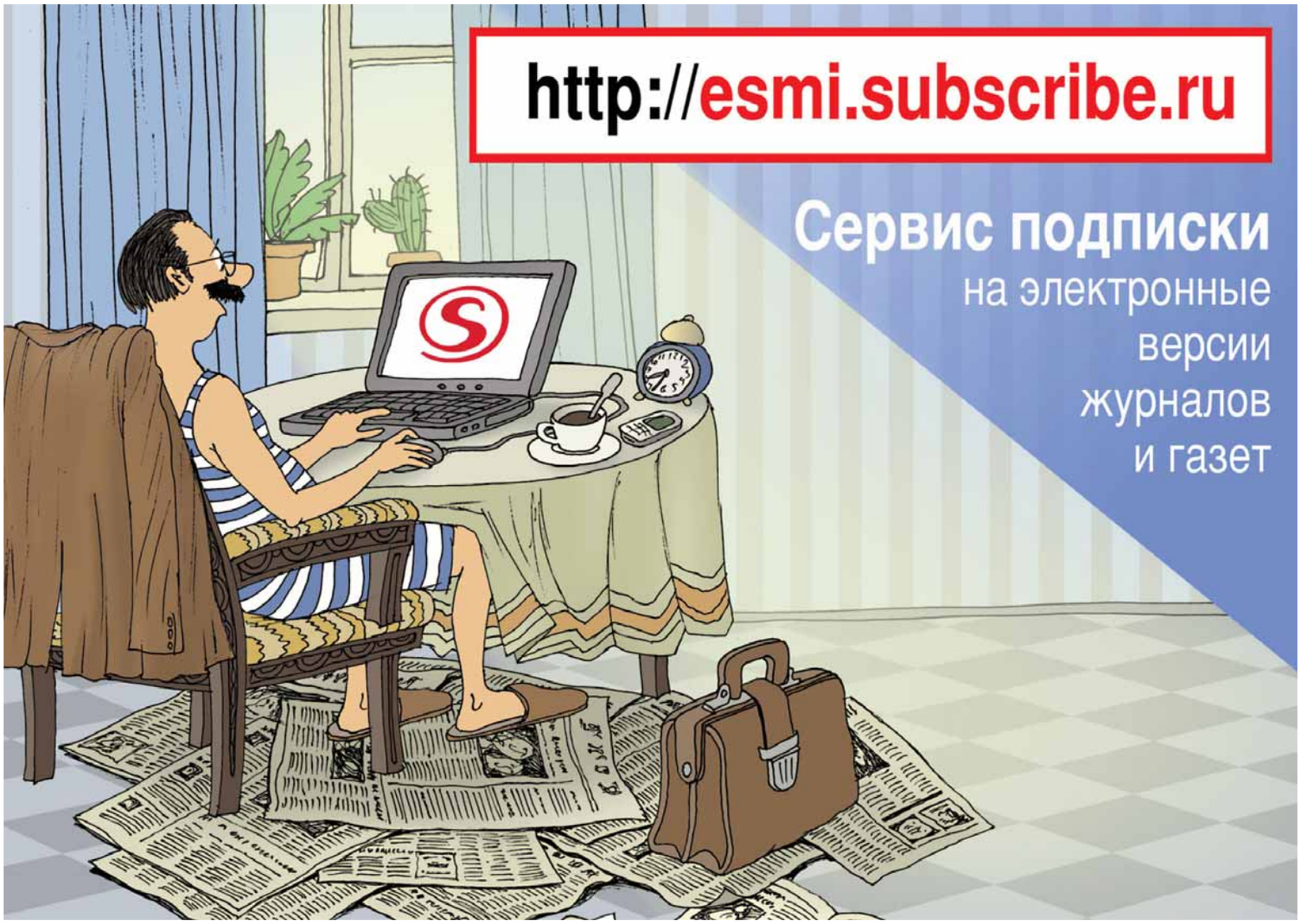
и

```
java -Djava.security.policy=open.policy RMIClient
```

Вот и все. Как можно заметить, чем проще и понятнее код клиента и сервера, тем больше телодвижений нужно сделать для запуска разных вспомогательных приложений. Это общая закономерность. Пока приложения простые, зачастую удобнее использовать просто сокеты; в более тяжелых случаях (например, для работы клиент-серверных приложений уровня предприятия, тех самых Enterprise Applications) проще один раз запустить что-то вроде *registry*, а потом пользоваться «благами цивилизации».

Стоит отметить, что RMI, создаваемый «по умолчанию», в нормальных приложениях использовать нельзя. Слишком много ограничений, слишком много неразрешимых проблем (например, если клиент и сервер находятся в разных подсетях, соединение становится почти невозможным; или – невозможно внести в *registry*-сервер, который запущен не на той же машине, где и *registry*). Но реализации, которые предоставляют так называемые контейнеры приложений (*GlassFish*, *Sun WebSphere*, *JBoss* и другие) – это именно то, что используется для связи клиентов с сервером. Там это все работает, и работает отлично. LXF

Через месяц Мы попробуем отправить побольше писем, не переходя границ дозволенного.



http://esmi.subscribe.ru

Сервис подписки
на электронные
версии
журналов
и газет

В «Системном администраторе» вы не прочтете о:

– котировках валют
– сплетнях
– погоде
– политике
– развлечениях

В вашем распоряжении:

– опыт лучших IT-специалистов
– новые идеи и полезные советы
– самые эффективные решения в области системного и сетевого администрирования

Подпишитесь сейчас!

Время подписки ограничено!

Роспечать – 20780, 81655
Пресса России – 87836
Online-подписка – www.linuxcenter.ru

Познакомьтесь

ЧАСТЬ 2 Сегодня **Александр Чернышов** намерен представить вам ризограф – аппарат, без которого ваша книга едва ли появится на свет, если вы печатаете внутривузовскую литературу (или что-то вроде этого).

В предыдущей части было уделено много внимания подготовке рисунков. Этот вопрос не относится к собственно вёрстке, но оказывается очень важным на этапе качественного воспроизведения результата на печатном оборудовании, поэтому его никак нельзя было проигнорировать.

Теперь, наконец, обсудим сами вопросы печати. На сегодняшний день при выходе «на станок» большинству из нас выбирать придётся из двух технологий: ризография или офсет.

Печать на ризографе

Ризография – самый распространённый сейчас способ так называемой «оперативной полиграфии». В силу распространённости таких аппаратов на сегодня это ещё и наиболее доступный способ печати небольших тиражей. Если же тираж предполагается большим (свыше 500 экземпляров) и надо «не очень срочно», имеет смысл подумать об офсете – «по деньгам» может выйти так же, а качество будет намного выше. Рассматривать физический смысл процесса ризографии не будем. Вместо этого поговорим о подготовке нашей вёрстки к тиражированию.

Обычно аппараты-ризографы печатают на бумаге привычных нам форматов A4, реже A3. В зависимости от возможностей переплётного оборудования типографии и принятой технологии одна страница итоговой книги может быть близка к формату A5, что обеспечивается складыванием листа формата A4 пополам и подборкой соответствующих листов в тетрадки (так и делают чаще всего!), может быть близка к формату A4 (A3 пополам), или даже A3 (но в этом случае шить можно только прокалывая толстой проволокой всю стопку листов в корешке.

Теоретически можно напечатать книгу и промежуточного формата (обратите внимание, что, например, большинство современных книг по программированию имеют формат страницы «посередине» между A5 и A4). Но в этом случае, во-первых, будет большой перерасход бумаги (её придётся просто обрезать с листов A3 после печати), а, во-вторых, не каждая такая типография пойдёт на изменение настройки своего переплётного оборудования. Короче, «промежуточные» форматы для ризографии неудобны.

Обычно, если вы не меняли размеры страницы в формате вёрстки *TeX*, они по умолчанию примерно соответствуют странице формата A5, но перед представлением своей работы в типографию лучше специально выяснить их требования. Вероятнее всего, вам выдадут требования для оригинал-макета, подготовленного на формате A4, который затем уже в самой типографии будет воспроизведён с уменьшением до A5. Требования сводятся к размеру полей. Особо надо оговорить расположение колонцифры (номера страницы). Договоритесь также о том, в каком виде должен быть представлен оригинал-макет – «на бумаге» (печатаете вы) или «на дискете» (печатает типография).

Масштабный коэффициент перевода из формата A5 в формат A4 примерно равен 1,44 или \magstep2 в нотации *TeX*. Пересчитав выданные значения полей в масштаб страницы, можно вычислить требуемый размер гранки – параметры \hsize и \vsize (например, для \hsize надо из ширины страницы вычесть ширины левого и правого полей). Полученные значения \hsize и \vsize следует задать в формате вашего документа и выполнить его трансляцию, после этого обязательно проверить результат и, при необходимости, выполнить ручную доводку вёрстки. Заметьте, что \hsize и \vsize задают именно гранку – без колонцифры. Требования типографии по колонцифре обычно можно проигнорировать. Результат чаще всего удовлетворит и вас, и типографию.

После получения окончательного DVI-файла, оригинал-макет можно подготовить к печати:

```
dvips -D 600 -t a4 -O 0.5cm,-0.2cm -x 1440 -o file.ps file.dvi
```

Параметры командной строки означают:

- -D – разрешение печати в dpi (точках на дюйм). В силу весьма среднего качества воспроизведения в ризографии разрешение больше 600 dpi бессмысленно. Есть один весьма тонкий момент, связанный с понятием жирности точки печати, но его мы обсудим чуть позже.
- -t – формат бумаги. Во многих системах *TeX* по умолчанию выбран Letter, так что осторожность не повредит;
- -O указывает смещение материала на странице по x и по y. Числа приведены только для примера. В каждом конкретном случае надо экспериментировать;
- -x указывает масштабный коэффициент, умноженный на 1000. В данном случае A5 масштабируется в A4 с коэффициентом 1,44;
- -o вывести результат в указанный файл (без этого ключа в большинстве систем Linux вывод пойдёт прямо на принтер).

Качество результата следует проконтролировать программой *gv*:

```
gv file.ps
```

Если всё устраивает, можно отправлять файл на печать. Если вы выполняете её сами, есть два варианта. Во-первых, можно найти

Месяц назад Мы научились готовить к печати иллюстрации на любой вкус.

с ризографом

принтер, непосредственно понимающий PostScript. На сегодня это не такая уж редкость (например, HP LaserJet 1200 можно найти во многих офисах). Важно убедиться, что его драйвер в системе настроен на PostScript, а не на PCL. Если принтер подключён к компьютеру с Linux и всё настроено правильно, то печать выполняется очень просто:

```
lpr file.ps
```

или даже

```
cat file.ps >/dev/lp0
```

если у вас есть соответствующие права записи в файл **/dev/lp0**. Подойдет и указанная выше команда dvips, если из командной строки убрать ключ -o.

Во-вторых, для печати на любом принтере можно воспользоваться *Ghostscript*. Простой, но неудобный способ – загрузить ваш **file.ps** в gv и печатать из меню. Впрочем, в большинстве современных установок Linux в подсистеме печати установлен набор фильтров, который задействует *Ghostscript* автоматически. Если это так, то для печати прекрасно будут работать описанные выше команды.

Но имейте в виду, что описанный выше ключ -O требуется подбирать под конкретный принтер, поэтому рекомендуется подготовить тестовую страницу, добиться её правильной печати, и только после этого печатать весь документ.

Наконец, если типография просит дать ей оригинал-макет в электронном виде, преобразуйте его в PDF:

```
ps2pdf file.ps file.pdf
```

и не забудьте внимательно просмотреть результат:

```
gv file.pdf
```

или

```
xpdf file.pdf
```

Почему так? Дело в том, что в 99 случаях из 100 все компьютеры типографии работают в Windows, который не поддерживает PS «из коробки». Покупать же дорогостоящие пакеты для работы с PS типографии незачем – аппараты-ризографы с ним не работают. Зато Adobe Reader или Foxit Reader для чтения и печати PDF там наверняка есть.

Принтер принтеру – не товарищ

Если типография сама печатает ваш оригинал-макет, то на качество оттисков вы влиять уже не можете. Если же вы печатаете оригинал-макет сами, то использование некачественных (перезаправленных) картриджей, а заодно и изношенных барабанов может привести к печальным последствиям для параметров используемых вами шрифтов. И если с изношенными барабанами всё более или менее ясно – будут появляться лишние чёрные точки и белые пропуски, то с картриджами проблема на самом деле носит более глобальный характер, поэтому рассмотрим её с общих позиций.

Найдите в вашей системе файл **modes.mf** и загляните в него. Вы увидите множество описаний принтеров, используемых программой *metafont* для растеризации шрифтов под конкретную модель. Обратите внимание: вам встретится несколько принтеров одного и того же разрешения (скажем, 600 dpi), но имеющих разные параметры генерации. Оказывается, с точки зрения полиграфии два принтера разных моделей с одинаковым разрешением печати могут оказаться совсем не эквивалентными! Их конструктивные особенности приводят к разной черноте (грубо говоря, размеру) печатаемых на бумаге точек. Программа *metafont*, генерируя шрифт для принтера, учитывает это обстоятельство, максимально сохраняя оптические (видимые глазом) характеристики шрифта.

Теперь становится понятно, что если вы замените принтер, не перегенерировав шрифт (хотя бы и с тем же разрешением), или поставите перезаправленный картридж (который уже не сможет обеспечить заводских параметров печати – скорее всего, она будет жирнее), итоговое качество печати «уплывёт».

Печать на ризографе можно рассматривать как вариант прямого оттиска, при котором тираж выполняется как копия с напечатанного на бумаге первого экземпляра. В некоторых случаях может оказаться, что оттиск получается некачественным (плохо пропечатывается и читается) из-за того, что «родные» гарнитуры *TeX* рассчитаны на высококачественное полиграфическое оборудование и тонковаты для снятия с них прямых оттисков. В этом случае надо специально получить гарнитуры с более жирным оттиском.

Интересно отметить, что именно в *TeX* (и, насколько мне известно, только в нём) эта задача может быть решена без собственно замены шрифтов, простой подстройкой системы *metafont*, растеризующей шрифты для конкретного принтера. Решение этой задачи довольно хорошо описано в статье А. А. Финогенова, доступной по адресу: **http://tex.imm.uran.ru/texserver/fonts/tipogr/**.

Вкратце: необходимо отредактировать файл **modes.mf**, создав в нем описание «своего» принтера по шаблону уже имеющегося, подобрать опытным путем параметр blacker (верхняя планка – 2.25), перегенерировать файл plain.base и заменить подстроку M ljfour в файле **config.ps** на M имя_своего_принтера.

Пожалуй, на сегодня достаточно. Оставляю вас подбирать правильное значение blacker, а в следующий раз мы завершим разговор о допечатной подготовке для *TeX*ников – и не только беседой об офсетной печати. LXF

Своенравный ps2pdf

В некоторых случаях простая команда типа

```
ps2pdf file.ps file.pdf
```

может привести к нежелательным результатам: некоторые страницы в **file.pdf** будут развёрнуты в горизонтальное положение. При печати вашей вёрстки «потоком» эти страницы так и пойдут на принтер развёрнутыми, поэтому нам надо обязательно получить все страницы в одинаковой ориентации.

К сожалению, документация по ключам команды *ps2pdf* распространяется только с исходными текстами системы *Ghostscript*. Правда, её можно найти и в Интернете (**http://ghostscript.com/doc/current**). Описанная выше проблема решается так:

```
ps2pdf -dAutoRotatePages=/None file.ps file.pdf
```

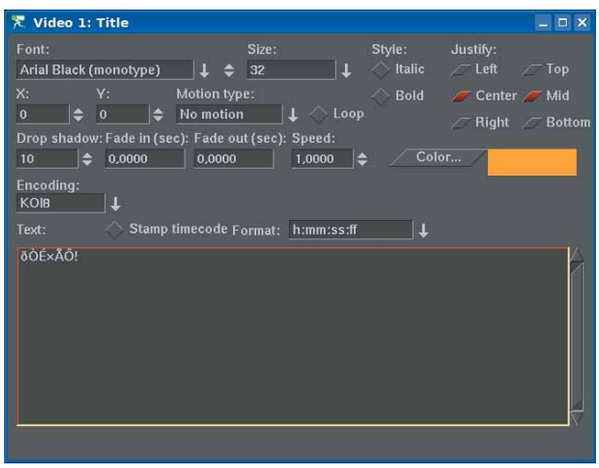
Через месяц Офсет не только для авторов бестселлеров – самый интересный способ напечатать ваш материал.

Титры

ЧАСТЬ 5 Любое кино заканчивается титрами, и **Петр Семилетов** завершает этот цикл статей разговором о титрах, о том как их делать и использовать в Cinelerra.

В *Cinelerra* есть встроенное средство для создания и редактирования титров. Технически титры воплощены в виде эффектов, так что расширение для работы с ними следует искать на странице Video Effects окна Resources.

На монтажном столе включите кнопку Arm на дорожке, куда хотите поместить титры, и кнопками [и] отметьте область, в которой хотите отображать их. Затем перетащите эффект Title в эту область. На дорожке, нажатием на кнопку с изображением увеличительного стекла, откройте редактор титров. Давайте посмотрим на него:



(Рис. 1).

В текстовом поле – там, где невразумительный набор символов, на самом деле написано по-русски слово «привет!». Вообще способ работы движка субтитров в *Cinelerra* с русским языком меня удивляет, так что расставим все точки над «i». Русские буквы не отображаются в редакторе, однако их вполне видно в окне Составителя (Compositor). Иными словами, в вашем видео русские титры все-таки будут. Однако, для этого в редакторе титров надо выбрать кодировку KOI8-R, даже если у вас, как у меня, установлена другая локаль. Когда вы сохраняете свой файл проекта, то титры сохраняются в нем же, причем в кодировке KOI8-R.

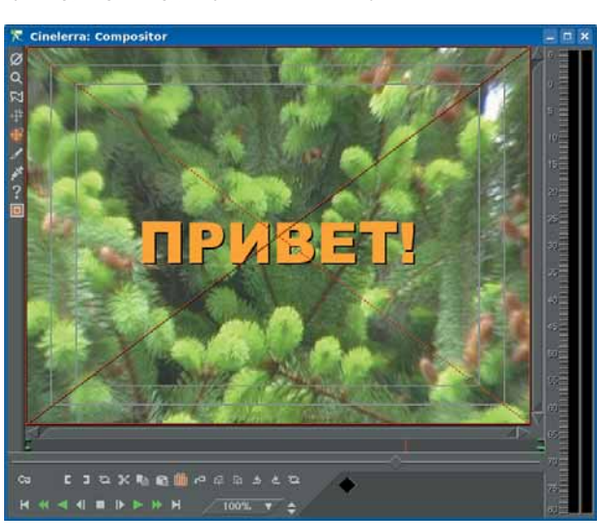
Если хотите редактировать титры с удобством, то закройте XML-файл проекта в *Cinelerra*, а затем просто откройте этот файл в любом текстовом редакторе, в кодировке KOI8-R. Найдите в нем раздел модуля титров. В нем, между тэгами <TITLE> и </TITLE>, и написан текст, который вы можете редактировать. Только не трогайте параметры, записанные для элемента TITLE (вроде TIMECODE) – их лучше менять через встроенный редактор в *Cinelerra*.

Познакомимся с ним ближе. Список выбора шрифтов предоставляет (по техническим причинам) доступ отнюдь не ко всем шрифтам, установленным в системе. *Cinelerra* берет шрифты (поддерживается формат TrueType) из каталога **/usr/lib/cinelerra/fonts**. Поэтому если вам нужен какой-то особый свой шрифт, поместите его туда, а затем дайте в этом каталоге команду:

```
ttmkfdir && mv fonts.scale fonts.dir
```

Далее, если отметить галочкой переключатель Stamp timecode, то вместо введенного вами текста титров будет отображаться временной код, формат которого можно задать тут же рядом, в выпадающим списке Format. По умолчанию там находится значение «h:mm:ss:ff», то где h – часы, mm – минуты, ss – секунды, ff – кадры.

Титры могут отбрасывать тень. Смещение текста относительно тени (именно так) задается параметром Drop shadow. Нельзя настроить направление тени – она отбрасывается только ниже и правее текста. При увеличении значения параметра Drop shadow текст смещается вверх, а тень ползет вниз и вправо. Тень эта черного цвета, без прозрачности. Думаю, что использовать ее на практике можно только когда ваши титры плохо видны на фоне видео – в этом случае установите небольшое значение тени: например, 1 или 2. Текст получит четкую границу, и буквы будет проще отличать от фона.



(Рис. 2).

Текст титров может иметь выравнивание (Justify) не только по вертикали, но и горизонтальное, в трех местах – Top (сверху), Mid (середина кадра), Bottom (внизу). Параметры X и Y задают смещение верхнего левого края титров от заданного выравниванием. Список Motion type (тип движения) определяет, как будут двигаться титры:

Месяц назад Клипы и сессии – искусство размещать материалы на единственном монтажном столе.

» No motion – титры статичны.
» Bottom to top – снизу вверх.
» Top to bottom – сверху вниз.
» Right to left – справа налево.
» Left to right – слева направо.

Переключатель Loop, будучи включенным, вызывает бесконечное воспроизведение титров, то есть крутит их по кругу. На скорость такой прокрутки влияет параметр Speed.

Fade in и Fade out – это плавное проявление титров и плавное же их затухание, обесцвечивание. Можно выбрать, сколько длится это проявление и затухание. Значение задается в секундах, по умолчанию равно нулю – титры показываются сразу же.

Весь этот арсенал средств позволяет делать как обычные «начальные» или «конечные» титры фильма, так и бегущую строку. Кроме того – вы, я думаю, это замечали во многих фильмах – титры бывают не только «бегущие» вертикально или горизонтально, но и покадровые. То есть, один кадр – режиссер Иванов. Через полторы секунды, другой кадр – оператор Сидоров. Между кадрами может идти переход – старый кадр угасает, новый проявляется. Как такое сделать в *Cinelerra*? Очень просто:

1 Создаем две видео-дорожки. На них располагаем через один кадры-титры. Каждый такой кадр – отдельный эффект Title, который длится нужное вам время.

Пару слов о расположении. Например, у нас есть кадры 1, 2, 3, 4, 5, 6. Кадры 1, 3, и 5 ставим на одной дорожке, кадры 2, 4, 6 – на другой, сохраняя при этом их последовательность, чтобы кадр 2 шел за кадром 1, только на другой дорожке. При этом конец каждого предыдущего кадра должен несколько задевать границу начала следующего (который находится на другой дорожке).

2 С помощью автоматизации кривой Fade (затухание) плавно делаем невидимым конец каждого кадра, и проявляем начало каждого кадра. Да, приходится делать это «вручную», поскольку использовать эффект-переход вроде Dissolve не получится: эффекты-переходы не влияют на титры. Однако обычные эффекты (Blur и тому подобные) – напротив, влияют. Напомню, что кривая автоматизации Fade окрашена белым. Если вы не видите ее (поначалу она вовсе не кривая, а просто ровная белая линия на дорожке), то включите ее в меню View > Fade, либо просто нажмите клавишу 3, находясь в окне монтажного стола.

В GIMP или в Cinelerra?

Встроенный редактор титров в *Cinelerra*, конечно, удобный инструмент, но всегда есть альтернатива. Можно ли изготовить текстовые титры во внешней программе, и есть ли предпосылки для этого? Предпосылки, несомненно, есть – если вам нужно нестандартное отображение текста. Предвижу также вопрос: «А как делаются анимированные титры, когда буквы весело вылетают из одной стороны экрана к другой, а затем выстраиваются в один ряд да еще подпрыгивают»? Увы, *Cinelerra* такого не умеет – во всяком случае, не автоматически. Нужно другое ПО – например, *Adobe After Effects*. Если вас это утешит, то *Adobe Premiere Pro* тоже не умеет делать такой эффект.

Ниже речь пойдет только о статичных титрах. Всегда можно сделать такие титры в *GIMP* или любом другом редакторе растровой графики. В KDE вот с некоторых пор появился отличный редактор *Krita*. Подойдет и векторный редактор, если в нем есть экспорт в PNG.

Рисуем титры на прозрачном фоне, затем экспортируем картинку в полноцветный PNG. Либо, если вы используете *Krita*, то можно в формат EXR, который тоже поддерживает альфа-канал, необходимый для обеспечения прозрачности фона. Обратите внимание, что в EXR на канал отведено 16 бит, а в PNG – 8. То есть один пиксел в EXR «весит» 64 бита, а в PNG – только 32 (количество каналов умножаем на разрядность). При восьми битах на канал в *Cinelerra* толку от EXR будет мало. Чтобы был толк, надо в свойствах проекта *Cinelerra* (меню Settings > Format) установить цветовую модель RGBA-Float. Но главное, обо всем этом не надо заботиться, если у вас не HDR-видео – видео с высоким динамическим диапазоном. Бытовые MiniDV-камеры на это не способны, не говоря уже о старых VHS.

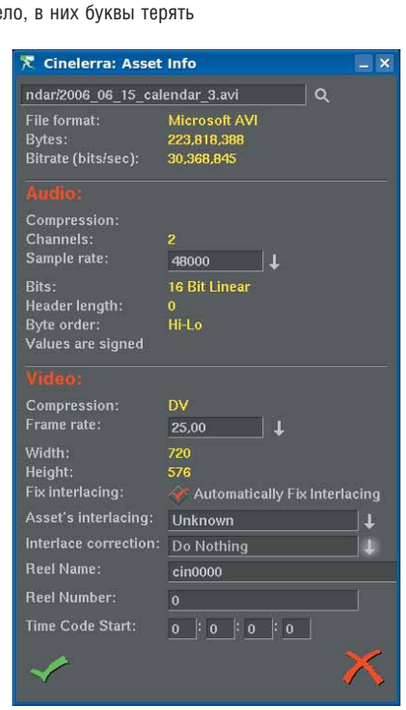
Каковы преимущества и недостатки исполнения титров в виде внешних файлов? Сначала скажем о недостатках. Несомненно, встроенный редактор титров позволяет упростить работу – а к тому же ускорить внесение изменений в текст (пусть даже при помощи внешнего текстового редактора). И напротив, с PNG-титрами надо все делать вручную. Вручную можно панорамировать вертикально или горизонтально многострочные титры. Вручную нужно располагать титры на экране. Но это более гибкий подход. Вы обращаетесь с такими титрами, как с обычными изображениями. А это значит, что можно использовать переходы (crossfades). Кроме того, в *GIMP* можно создать куда более художественные изображения текста, нежели позволяет обыкновенный шрифт, пусть даже и сдобренный тенью. Конечно, «художественные» титры не всегда нужны, да и хороши они тогда, когда толково сделаны. Лучшие примеры тому смотрите в каких-нибудь советских старых сказочных фильмах или мультиках. Вот где истинно художественные титры!

Что до пресловутых анимированных буковок, то их в *Cinelerra* сделать все-таки можно. Но этот труд под силу только очень кропотливому человеку. Создаете на каждую букву по PNG-файлу, и при помощи автоматизации проектора перемещаете буквы в нужном вам порядке и с должной скоростью. Думаю, терпения у вас хватит разве что на заглавие фильма или на слово «Конец», потому что составлять таким образом все титры – задача слишком трудоемкая.

И напоследок...

В завершение дам еще один совет по титрам, который, впрочем, годится для любой развитой программы видеомонтажа. Да, смонтированное вами видео будет воспроизводиться на компьютере в том же виде, как вы его смонтировали. Однако при просмотре на бытовой технике, например, на DVD-проигрывателе, подключенном к телевизору, часть изображения по периметру кадра будет обрезана. На современных здоровенных и дорогих телевизорах такого может не произойти, однако на более старых моделях это вполне вероятно.

Поэтому, в окне Compositor следует включить кнопку Show safe regions («Показывать безопасные области»). Появится две рамки, внутренняя и внешняя. Следите, чтобы титры не выползали за пределы внутренней рамки, а значимое видео – за пределы внешней. Повторюсь: это в случае, если вы хотите, чтобы ваше видео полностью помещалось на экране среднестатистического телевизора. Насчет внешней рамки можете не беспокоиться. Ну, не увидят на телевизоре какой-то части по периметру, однако зрителю все равно будет понятно происходящее на экране. Титры же – другое дело, в них буквы терять нельзя. Поэтому хорошее правило – всегда помещать титры в пределах внутреннего прямоугольника. LXF



» Рис. 3.

WPA Enterprise

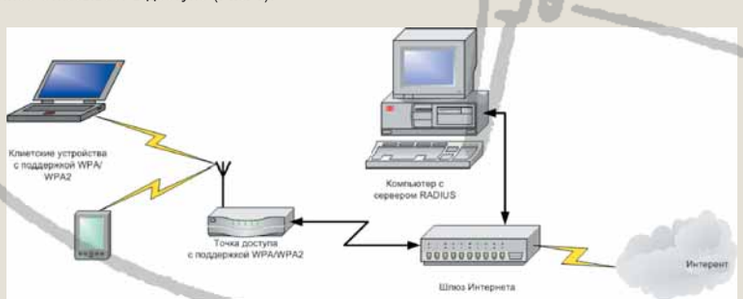
ЧАСТЬ 1 Ломаете голову, чем бы заняться дождливым осенним вечером? **Андрей Боровский** припас для вас учебник по укреплению безопасности домашней сети.

Как появляется домашняя компьютерная сеть? Сначала у вас дома только один компьютер, затем вы покупаете новый. Старую машинку выбрасывать жалко, продать – некому. Вы связываете два компьютера витой парой. Затем вы покупаете ноутбук. Помимо всего прочего, ноутбук – очень подходящее устройство для того, чтобы бродить по всемирной паутине, лежа на диване, или слушать интернет-радио, сидя на балконе. Однако таскать витую пару вслед за ноутбуком неудобно, и поэтому вы устанавливаете беспроводную сеть. В процессе установки беспроводной сети у себя дома большинство пользователей идет по пути наименьшего сопротивления, что, увы, делает их творения крайне небезопасными. Статистические исследования я не проводил, но сканирование эфира в доме, где я живу, выявило три Wi-Fi сети (помимо моей собственной), две из которых вообще не были защищены. Об уязвимости простого протокола защиты WEP было сказано столько, что распространяться на эту тему представляется излишним. Необходимый и достаточный уровень защиты домашних беспроводных сетей может быть обеспечен протоколами WPA/WPA2 с разделяемым ключом (pre-shared key, PSK). Настроить этот протокол не намного сложнее, чем обычный WEP. Недостатки WPA-PSK, главный из которых – использование одного и того же ключа на всех клиентских машинах, в домашней сети несущественны. Однако, хотя WPA-PSK является вполне надежным методом защиты небольшой беспроводной сети, среди пользователей Linux наверняка найдутся такие, которые, как и я, не захотят останавливаться на достигнутом. Ведь каждый домашний пользователь Linux, в некотором смысле, исследователь. Практически каждый стремится выйти за рамки простейшей технологии, ориентированной на «чайников», и опробовать другие, предоставляющие более широкие возможности. Сегодня мы поговорим о том, каким образом домашний Linux-сервер может расширить возможности вашей беспроводной сети с помощью сервера *FreeRADIUS*, который является наиболее популярной на сегодняшний день реализацией сервера протокола RADIUS.

Терминология

Под аутентификацией понимается процесс, который подтверждает, что пользователь, пытающийся зарегистрироваться в системе, действительно тот, за кого он себя выдает. Авторизация – это процесс, подтверждающий определенные права пользователя, прошедшего аутентификацию. Устройство доступа к сети, реализующее физический уровень системы аутентификации, часто называют аутентификатором.

Как работает сервер RADIUS? Для того чтобы иметь возможность авторизовать пользователей в сети с помощью RADIUS, необходимо чтобы аутентификатор (в случае беспроводной сети – точка доступа) поддерживал протокол RADIUS. Получив запрос на авторизацию, RADIUS-аутентификатор направляет запрос RADIUS-серверу, используя, как ни странно, протокол RADIUS. В зависимости от ответа сервера аутентификатор либо предоставляет пользователю доступ к сети, либо отказывает в доступе (**Рис. 1**).



Радиус чего?

Аббревиатура RADIUS расшифровывается как Remote Authentication Dial-In User Service [Служба удаленной аутентификации входящих пользователей]. Словосочетание Dial-In не должно вводить вас в заблуждение. Протокол RADIUS используется для авторизации, аутентификации и учета пользователей в самых разных системах (в том числе и беспроводных). Для того, чтобы понять, нужен ли вам RADIUS-сервер и как его использовать, вы должны понимать место протокола RADIUS в системе аутентификации и авторизации. Протокол RADIUS полностью абстрагирован не только от технологии связи, но и от протоколов передачи данных в сети, в которой происходит авторизация. Фактически функция сервера RADIUS сводится к тому, что сервер принимает информацию, которую пользователь сети предоставляет в процессе аутентификации, и авторизует пользователя. Это означает, что использование сервера RADIUS не может сделать безопаснее сам процесс передачи данных в сети (оно может обезопасить только процесс передачи данных в ходе аутентификации, что, конечно, тоже немаловажно). После того как процесс авторизации пользователя в сети завершился, сервер RADIUS ему больше не нужен, по крайней мере, до тех пор, пока пользователю не понадобится снова авторизоваться в системе.

Каковы преимущества авторизации с помощью RADIUS перед авторизацией с помощью предустановленного ключа? Прежде всего, использование RADIUS позволяет дифференцировать пользователей. Предустановленный ключ у всех пользователей общий, а RADIUS позволяет выдать каждому пользователю персональное имя учетной записи и пароль (или персональный сертификат). Второе преимущество RADIUS заключается в возможности вести учет сетевой активности пользователей. Беспроводные сети активно замещают собой не только последнюю милю, по которой трафик подается конечным пользователям, но и витую пару, которую жильцы дома могут проложить между квартирами. Впрочем, забудьте, что я это говорил...

Протокол RADIUS применяется уже довольно давно. Несколько

НА ДОМУ

лет назад строителям беспроводных сетей был предложен протокол DIAMETER. Этот протокол, как можно догадаться из его названия, представляет собой улучшенную версию RADIUS. Впрочем, не смотря на готовую замену, RADIUS по-прежнему процветает, а дополнительные возможности DIAMETER вряд ли понадобятся администраторам небольших сетей, да и далеко не все сетевое оборудование, доступное на сегодняшний день, поддерживает эти возможности.

О достоинствах RADIUS было сказано выше, главный же недостаток системы авторизации с помощью RADIUS в домашней сети очевиден: для того, чтобы пользователи могли авторизоваться, RADIUS-сервер должен работать, то есть один из компьютеров в сети должен быть постоянно включен, что может показаться кому-то избыточным. Стоит отметить, что некоторые маршрутизаторы, совмещенные с точками беспроводного доступа, оснащены встроенным сервером *FreeRADIUS* (или позволяют установить его), однако в домашних сетях такие устройства редки.

Не следует, однако, думать, что *FreeRADIUS* – редкий гость в малых сетях. По данным разработчиков, сервера сети с небольшим числом узлов (до 10) составляют около 14% от общего числа сетей, использующих *FreeRADIUS,* и это не мало, если учесть широкий спектр применения сервера (в том числе, в сетях с десятками и даже сотнями тысяч узлов).

Враг не пройдет!

Каковы же дополнительные возможности аутентификации, предоставляемые сервером RADIUS? В настройках клиентского ПО беспроводной сети вы можете встретить аббревиатуры EAP, EAP-TLS, PEAP-MSCHAP-V2 и другие. Они обозначают различные протоколы (а иногда и семейства протоколов) аутентификации. Так, EAP (Extensible Authentication Protocol, Расширяемый протокол аутентификации) представляет собой контейнер для протоколов, фактически осуществляющих передачу запросов на аутентификацию и ответов аутентификатора. Поскольку EAP абстрагирован от конкретных процедур аутентификации, сетевое оборудование, поддерживающее EAP, тоже не связано с какими-либо конкретными протоколами (в частности, это означает, что оборудование, поддерживающее WPA и WPA 2, может работать с одними и теми же протоколами аутентификации). Новые протоколы аутентификации могут свободно применяться на старом оборудовании.

Компания Microsoft разработала на основе EAP протокол EAP-TLS, принятый позднее в качестве стандарта. Протокол EAP-TLS (Extensible Authentication Protocol – Transport Layer Security) реализует двустороннюю аутентификацию с использованием сертификатов X.509 обеими сторонами. PEAP (Protected Extensible Authentication Protocol) представляет собой проприетарное расширение протокола EAP. В отличие от EAP, протокол PEAP создает безопасный туннель перед началом процесса аутентификации. Далее по этому туннелю передаются данные для аутентификации. Наличие безопасного туннеля затрудняет перехват данных злоумышленником. Как и EAP, PEAP представляет собой контейнер для высокоуровневых протоколов аутентификации, например, для MSCHAP, в которых аутентификация выполняется с помощью имени учетной записи и пароля (стоит отметить, что изначально PEAP предназначался для более продвинутых протоколов аутентификации, нежели MSCHAP, но в настоящее время устройства доступа поддерживают PEAP-MSCHAP V2). Еще один проприетарный протокол, EAP-TTLS, также использует безопасные туннели для передачи данных. EAP-TTLS интересен тем, что рассматривается IETF как будущий стандарт безопасной аутентификации. Как и в EAP-TLS, в EAP-TTLS реализована двусторонняя аутентификация (сервер удостоверяет себя с помощью сертификата X.509). Мы же рассмотрим аутентификацию с помощью PEAP-MSCHAP V2 и EAP-TLS.

Скорее всего, ваш дистрибутив Linux содержит пакет *FreeRADIUS*. Последнюю версию сервера (а при работе с программами, связанными с безопасностью, всегда следует использовать последнюю версию) можно загрузить с сайта проекта **www.freeradius.org**. В этой статье рассматривается новейшая на данный момент версия *FreeRADIUS 1.1.7*, установленная из исходных текстов.

Установим FreeRADIUS

По умолчанию в системе Linux исполняемый файл *FreeRADIUS* (*RADIUSd*) устанавливается в директорию **/usr/local/sbin**. Директория **/usr/local/etc/raddb/** содержит файлы конфигурации сервера.

Основные настройки сервера содержатся в файле **radiusd.conf**. Домашнему сетестроителю, скорее всего, не придется вносить изменения в этот файл. Если же вы хотите, чтобы сервер RADIUS был связан только с одним IP-адресом (по умолчанию сервер регистрируется на всех IP той машины, на которой он запущен) или прослушивал нестандартный порт, подробные комментарии в файле подскажут вам, как его редактировать. Справку о структуре файла **radiusd.conf** можно также получить по команде

```
man 5 radiusd.conf
```

В процессе настройки сервера я рекомендую запускать его с ключами -fx. Ключ -f позволяет запустить программу в терминале (а не в виде демона), а ключ -x включает режим частичного отображения отладочных сообщений, которые помогут вам найти и исправить неизбежные ошибки, возникающие в процессе настройки сервера. Если во время запуска сервера не возникло никаких проблем, серия сообщений, выводимых сервером в режиме отладки должна закончиться строками

```
Listening on authentication *:1812
Listening on accounting *:1813
Ready to process requests.
```

Порт 1812 по умолчанию принимает запросы от аутентификатора, а порт 1813 используется для обработки учетной информации о пользователях сети. После этого сервер переходит в режим ожидания запросов на аутентификацию. Теперь, когда вы убедились, что сервер запускается, вы можете приступить к настройке взаимодействия сервера и точки доступа. Для этого ее адрес, а также некоторые другие параметры, должны быть прописаны в файле **clients.conf** (название файла объясняется тем, что для сервера RADIUS точки доступа выступают в роли клиентов).

Сервер RADIUS может обслуживать несколько аутентификаторов одновременно, и каждому из них в файле **cliens.conf** должна соответствовать одна запись вида:

```
client 192.168.1.50/24 {
        secret      = ...
        shortname = private-network-1
        nastype = other
}
```

Здесь 192.168.1.50/24 – адрес аутентификатора в сети, параметру secret должен быть присвоен секретный ключ аутентификатора. Этот ключ используется для шифрования и подписи сообщений, которые аутентификатор посылает серверу RADIUS. Параметр shortname содержит имя узла точки доступа. Для каждого аутентификатора можно также указать его тип (параметр nastype). Если вашей модели аутентификатора нет в списке моделей, известных серверу, присвойте этому параметру значение other, или просто удалите его из записи. Более подробные сведения о файле **clients.conf** вам даст вам команда

```
man 5 clients
```

После настройки файла **clients.conf** следует настроить саму точку доступа. Обычно в настройках точки доступа указываются адрес и порт сервера, а также секретный ключ – тот же, что и в параметре secret.

Теперь вы можете проверить, обращается ли точка доступа к серверу RADIUS. При попытки авторизации беспроводного клиента в сети в окне терминала, в котором запущен сервер (напомню, что мы запускаем его с ключами -fx), должна появиться серия сообщений вида

```
rad_recv: Access-Request packet from host 192.168.1.50:1812, id=7,
length=197
Sending Access-Reject of id 6 to 192.168.1.50 port 1812
```

Сервер получил запрос от точки доступа, но отказал пользователю в авторизации, поскольку требуемой учетной записи еще не существует.

Однако прежде чем перейти к созданию учетной записи пользователя на сервере рассмотрим, еще один важный файл настроек. Сервер *FreeRADIUS* имеет модульную структуру, и для того, чтобы он мог обрабатывать запросы по тому или иному протоколу, сервер должен загрузить модуль, в котором реализована поддержка данного протокола. Для наших экспериментов нам понадобятся модули *eap*, *mschapv2*, *tls*, *ttls* и *peap* (их названия соответствуют названиям протоколов аутентификации, которые они поддерживают). Загрузкой модулей протоколов и их настройкой управляет файл **eap.conf**. Чтобы сервер загружал необходимые модули, нам нужно раскомментировать соответствующие им записи в **eap.conf**. Они имеют следующий вид

```
имя_модуля {
        параметр = значение
        ...
}
```

Нам пока не требуется изменять значения параметров модулей – достаточно просто убрать значки комментариев (модули *eap* и *mschapv2* загружаются по умолчанию, так что их записи в файле **eap.conf** действительны).

Настроим: PEAP MS CHAP V2

Данные учетных записей пользователей для аутентификации по протоколу PEAP MS CHAP V2 хранятся в файле users (а не **users.conf**, как можно было бы ожидать). Подробную информацию об этом файле вы можете получить с помощью команды

```
man 5 users
```

Сервер *FreeRADIUS* может хранить данные учетных записей не только в файле **users**, но и в базах данных на серверах *MySQL*, *SQL Server*, *Oracle*, однако для домашней сети использование сервера БД для хранения данных о пользователях – пустое расточительство вычислительных ресурсов, и мы его рассматривать не будем. Предполагается также, что в вашей домашней сети отсутствуют контроллеры доменов. Кроме того, для простоты изложения мы присвоим клиентской машине статический адрес.

При настройке учетной записи пользователя, проходящего аутентификацию по протоколу MS-CHAP V2, в файл **users** следует добавить запись вида:

```
user1       User-Password = "**********"
            Service-Type = Framed-User,
            Framed-Protocol = SLIP,
            Framed-IP-Address = 192.168.1.3,
            Framed-IP-Netmask = 255.255.255.0,
            Framed-Routing = Broadcast-Listen,
            Framed-Filter-Id = "std.ppp",
            Framed-MTU = 1500,
            Framed-Compression = Van-Jacobsen-TCP-IP
```

Этот фрагмент текста описывает учетную запись пользователя с именем **user1**. Атрибут User-Password содержит пароль (который должен быть заключен в кавычки). Атрибут Framed-Protocol позволяет указать протокол, используемый для передачи данных между клиентом и сетью. Возможные значения этого атрибута – SLIP и PPP. Протокол PPP обладает более широкими возможностями, но для домашней сети нам вполне достаточно SLIP. Атрибуты Framed-IP-Address и Framed-IP-Netmask указывают адрес нового узла сети и маску подсети. Атрибут Framed-MTU позволяет указать значения MTU. Последние три атрибута должны соответствовать настройкам клиентской машины. После того как вы перезапустите сервер и настроите протокол MS-CHAP V2 на клиентской машине, ваш компьютер может регистрироваться в беспроводной сети с помощью имени пользователя и пароля.

Настроим: EAP-TLS

Настройка аутентификации с помощью сертификатов X.509 требует больших усилий. Прежде всего, нам следует создать сами сертификаты, для чего мы воспользуемся утилитой *openssl*. Если для обмена почтой по протоколу SMIME нужен сертификат, выданный общепризнанным удостоверяющим центром, то для аутентификации в беспроводных сетях гораздо удобнее создать собственный корневой сертификат и с его помощью подписывать сертификаты сервера и клиентов. Прежде всего нам нужно сгенерировать 2048-битный ключ для корневого сертификата:

```
openssl genrsa -des3 -out myroot_ca.key 2048
```

Ключ будет сохранен в файле **myroot_ca.key**. Теперь создадим сам сертификат:

```
openssl req -new -x509 -days 730 -key myroot_ca.key -out myroot_ca.crt
```

Сертификат, действительный в течение 730 дней, сохраняется в файле **myroot_ca.crt**. В процессе генерации сертификата программа задаст вам несколько вопросов относительно значений полей сертификата.

Для генерации серверного и клиентского сертификатов, подписанных электронной подписью, заверенной нашим корневым сертификатом, нам понадобится создать специальный файл конфигурации, который будет использовать утилита *openssl*. Пример файла конфигурации вы найдете на диске (он называется **ca.config**), приводить его полный текст здесь я не буду. Файл конфигурации содержит ответы на стандартные вопросы, которые задает утилита *openssl* в процессе генерации сертификата (вы, конечно, захотите заменить эти данные своими собственными). Кроме того, этот файл устанавливает пути к некоторым каталогам, используемым в процессе генерации сертификатов и некоторые дополнительные параметры. Строки

```
[ xpclient_ext ]
extendedKeyUsage = 1.3.6.1.5.5.7.3.2
[ xpserver_ext ]
extendedKeyUsage = 1.3.6.1.5.5.7.3.1
```

указывают расширенные параметры сертификата. Они понадобятся в том случае, если вы генерируете сертификаты (клиентский и серверный) для сети, в которой предусмотрена авторизация Windows-клиентов. Файл **ca.config** ссылается на файлы и директории, который долж-

ны быть созданы явным образом (приведенная ниже последовательность команд предполагает, что имена файлов и директорий в **ca.conf** не были изменены). Сами команды выполняются в той директории, в которой мы сохранили корневой сертификат и его ключ.

```
mkdir newcerts
touch certs.db.index
echo “01” > certs.db.serial
```

Теперь мы можем приступить к созданию сертификата сервера. Ключ для него получается уже знакомой нам командой:

```
openssl genrsa -des3 -out myserver.key 2048
```

Далее мы создаем запрос на подпись сертификата точно так же, как если бы мы обращались к настоящему УЦ:

```
openssl req -new -key myserver.key -out myserver.csr
```

На основе этого запроса мы создаем серверный сертификат с подписью, заверенной корневым сертификатом:

```
openssl ca -config ca.config -in myserver.csr -out myserver.crt –batch
```

Сертификат клиента создается по той же схеме:

```
openssl genrsa -des3 -out user1.key 2048
openssl req -new -key user1.key -out user1.csr
openssl ca -config ca.config -in user1.csr -out user1.crt -batch
```

Важный момент: имя сертификата клиента (параметр CN) не должно совпадать с именем сертификата сервера и корневого сертификата. Для переноса на платформу Windows сертификат **user1.csr** следует экспортировать в формате PKCS#12:

```
openssl pkcs12 -export -in user1.crt -inkey user1.key -out user1.p12
```

Теперь мы переходим к собственно настройке *FreeRADIUS*. В директории **/usr/local/etc/raddb/certs** необходимо создать файл с числами Диффи-Хеллмана (Diffie-Hellmann):

```
openssl dhparam 512 > /usr/local/etc/raddb/certs/dh
```

Там же создаем файл случайной последовательности из 1024 байтов:

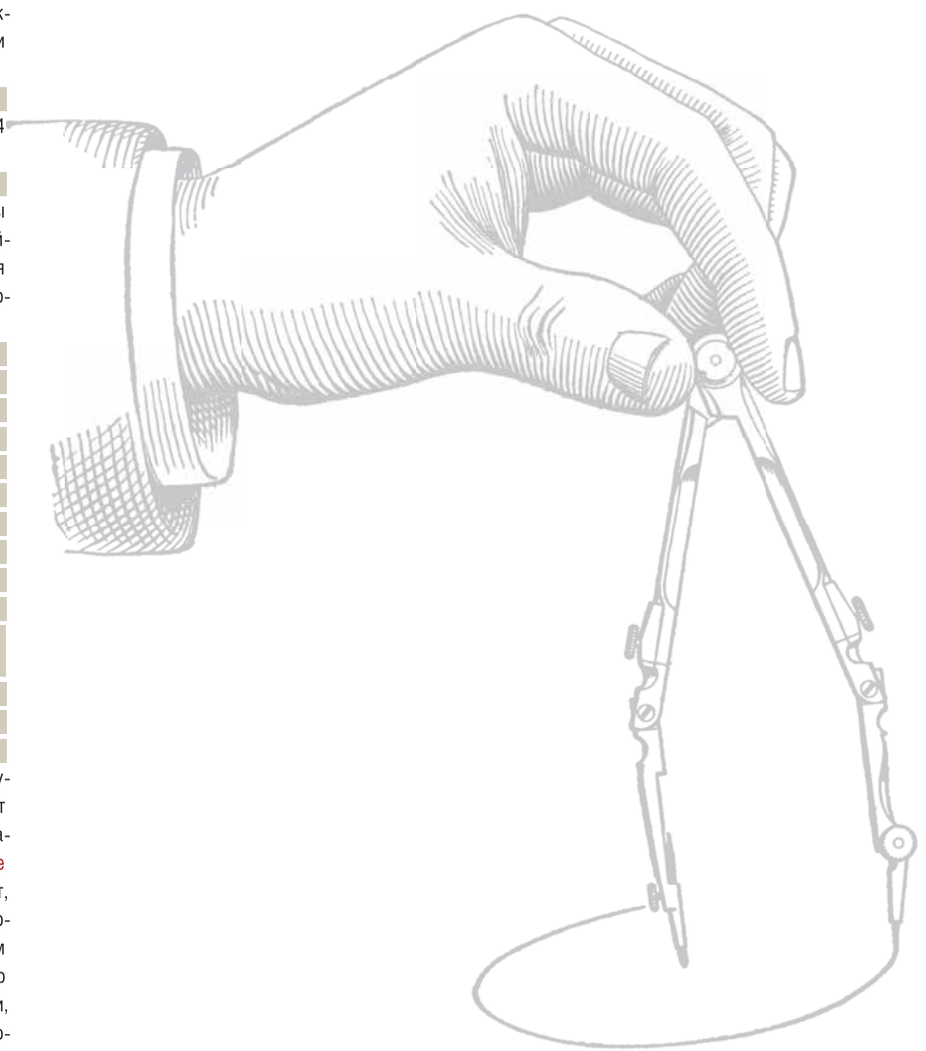
```
dd if=/dev/urandom of=/usr/local/etc/raddb/certs/random count=2
```

Теперь скопируем в директорию **/usr/local/etc/raddb/certs** файлы **myroot_ca.crt**, **myserver.crt** и **myserver.key**. Нам осталось настроить файлы конфигурации *FreeRADIUS*, а точнее **eap.conf**. После редактирования раздел параметров модуля *tls* в этом файле должен выглядеть примерно так (ради сокращения листинга комментарии пропущены):

```
tls {
        private_key_password = ******
        private_key_file = ${raddbdir}/certs/myserver.key
        certificate_file = ${raddbdir}/certs/myserver.crt
        CA_file = ${raddbdir}/certs/myroot_ca.crt
        dh_file = ${raddbdir}/certs/dh
        random_file = ${raddbdir}/certs/random
        fragment_size = 1024
        include_length = yes
        check_crl = no
        check_cert_issuer = “/C=RU/ST=MOSCOW/L=Moscow/O=Inet
Ltd/CN=Andrei Borovsky/emailAddress=anb@symmetrica.net”
        #check_cert_cn = %{User-Name}
        cipher_list = “DEFAULT”
        }
```

Атрибут private_key_password должен содержать пароль доступа к секретному ключу серверного сертификата (вы указывали этот пароль, когда создавали соответствующий ключ). Назначение параметров certificate_file, CA_file – на файл корневого сертификата, dh_file и random_file должно быть очевидно. Атрибут check_crl указывает, должен ли сервер сверять сертификат клиента со списком отзыва сертификатов. Мы не создавали список отзыва, поэтому устанавливаем значение по. Атрибут check_cert_issuer определяет, должен ли сервер сверять данные об организации-эмитенте сертификатов с данными, записанными в клиентском сертификате. Вы можете закомментировать строку check_cert_issuer, если не хотите выполнять эту проверку, в противном случае сюда следует вписать данные эмитента клиентского сертификата. Атрибут check_cert_cn указывает шаблон имени пользователя, с которым следует сверять значение CN клиентского сертификата. Следует иметь в виду, что система Windows обычно добавляет к имени пользователя разного рода вспомогательную информацию. Мы можем записать в атрибут check_cert_cn шаблон, позволяющий извлечь настоящее имя пользователя из строки, переданной Windows, а можем отменить эту проверку, закомментировав строку (что мы и делаем). На этом настройка сервера закончена. Нам остается установить корневой и клиентский сертификаты на клиентских машинах (мы не рассматриваем процесс настройки Windows-клиентов; если вы хотите получить подробные инструкции, обратитесь, например, к статье **http://www.ixbt.com/comm/prac-wpa-eap.shtml**, которая содержит пошаговые инструкции настройки Windows, но несколько устарела в том, что касается *FreeRADIUS*).

В чем же смысл аутентификации EAP-TLS? Фактически сервер проверяет только то, заверена ли подпись клиентского сертификата известным ему корневым сертификатом. Иначе говоря, для предоставления пользователю доступа к сети достаточно выдать пользователю сертификат, подпись которого удостоверена корневым сертификатом, известным *FreeRADIUS*. Ну, а поскольку подделать эту подпись достаточно тяжело, злоумышленники, желающие посидеть в Интернете за чужой счет, будут держаться подальше от вашей сети! LXF



» **Через месяц** Мы разберемся в работе беспроводных клиентов, существующих для Linux.

Ответы

Есть вопрос по Open Source? Пишите нам по адресу: **answers@linuxformat.ru**

Сегодня мы ответим на вопросы о:

1. Удаленную печать
2. gHamachi
3. Vista
4. Группы мониторов
5. Samba
6. BT Wi-Fi
7. Сетевые Карты
8. Детали ядра
9. Поиску «нюхачей»
10. Русификации Puppy
11. Видео на PSP
- ★ Linux в домене
- ★ Apache

1 Удаленная печать

Я учитель в школе, и когда я занялся компьютерами в учительской, то на них всех стоял Windows. Сейчас я готов установить Ubuntu на один из них, но это осложняется одной «мелкой» деталью – принтером!

Имеется компьютер с Windows Server, подсоединенный к коммутатору. Сервер работает только как принт-сервер. Я запускаю Ubuntu Feisty Fawn, но не могу печатать. Ubuntu находит принтер, Samsung CLP-500, я устанавливаю драйвера, но ничего не печатается. Может, надо использовать *Samba*?

Эдуардо Рамальядейро [Eduardo Ramalhadeiro]

Согласно базе OpenPrinting (**www.linuxfoundation.org/en/OpenPrinting**), этому принтеру необходимы драйверы *SplitX*, их можно найти на сайте **http://splix.sourceforge.net**. Хотя этот драйвер хорошо работает с некоторыми лазерными принтерами Samsung (например, мой монохромный лазерник работает отлично), для CLP-500 он помечен как работающий «частично». Это, по-видимому, связано с ограничением разрешения до 600 dpi при печати. Samsung также предлагает Linux-драйвера, которые можно загрузить с **http://short.zen.co.uk/?id=792** (полный URL до смешного длинный).

› Установив пакет *SplitX*, вы получите работающие открытые драйвера для принтеров Samsung.

SplitX включен в состав текущей версии Ubuntu, так что весь вопрос заключается в его установке через *Synaptic* и выборе нужного драйвера в меню настройки принтера.

CUPS может «разговаривать» с Windows-принтерами – он использует клиентские библиотеки *Samba*, так что Вам нужна установленная *Samba*, но самому ее настраивать не надо: Ubuntu устанавливает *Samba* по умолчанию, и незачем совершать лишние движения. Все, что нужно сделать – это установить пакет *SplitX* через *Synaptic*, а затем запустить Новый принтер в Система > Администрирование > Печать и, когда спросят, выбрать корректный принтер. **НБ**

2 gHamachi gхамит

Пытаюсь настроить *gHamachi*, используя уроки вашего июньского номера [LXF93]. Я следовал инструкциям на стр. 63 до этапа 3, когда я нажал Yes и ввел пароль root. Тут появилось сообщение: «TAP/TUN NOT FOUND». У меня Ubuntu 6.06. В чем может быть проблема?

Брайан Митчелл [Bryan Mitchell]

Эта проблема вызвана тем, что *gHamachi* не находит программу **tuncfg**, которая происходит от *Hamachi*, так что придется установить также и *Hamachi*. Сообщение об ошибке довольно-таки дезинформирующее: TAP/TUN не найден, потому что нет программы, используемой для его поиска. Похоже, *gHamachi* выдает одно и то же при любых ошибках запуска **tuncfg**, даже если попытка сорвалась из-за того, что **tuncfg** не найден.

Вам понадобится несколько дополнительных команд для установки *Hamachi*. Зайдите в *Synaptic* и установите пакет build-essential, содержащий все необходимое для установки пакетов вне *Synaptic*. Теперь загрузите Linux-версию *Hamachi* с **http://hamachi.cc**.

Если вы сохранили ее на вашем рабочем столе (Desktop), откройте терминал и наберите:

```
cd Desktop
tar -xf hamachi-0.9.9.9-20-lnx.tar.gz
cd hamachi-0.9.9.9-20-lnx
sudo make install
```

Теперь Вы готовы запустить *gHamachi* и продолжить действия по инструкциям. **НВ**

3 Визит к Vista

После проблем с Vista друг попросил меня установит на его компьютере Linux. На моем компьютере – FedoraCore 6, и я открыл для совместного доступа диск в Vista, чтобы сохранить необходимые файлы друга. Но мне нужна помощь в переносе файлов. Я могу получить доступ к общему диску, но когда я пытаюсь открыть папки, чтобы »

Наши эксперты

» Мы найдем эксперта по любому вопросу. Вы получите ответ на все: от проблем с установкой или модемом до сетевого администрирования; главное – спросить!

Нейл Ботвик
Поруководив ISP и передактировав диски для нашего журнала, Нейл считает, что в Linux он от скуки на все руки.

Майк Сондерс
Майк был одним из создателей прототипа LXF – Linux Answers. Его специальности – программирование, оконные менеджеры, скрипты инициализации и SNES.

Пол Хадсон
Пол – местный супер-программист. А заодно он занимается всеми вашими проблемами по части web и баз данных.

Ник Вейч
В свободное от исчеркивания текстов красными чернилами время Ник возится с Linux- графикой и 3D-приложениями; он у нас отвечает за простые вопросы!

Владимир Иванов
Владимир работает с Linux уже более 10 лет, днем – создавая комплексные решения для крупных компаний, а вечером – модерируя сайт linux.org.ru и отвечая на вопросы о системном администрировании и информационной безопасности.

Сергей Супрунов
Сергей считает, что лучший отдых – перемена занятия, поэтому отдыхает от FreeBSD-сервера и вопросов пользователей, копаясь в Linux и заглядывая на форумы.

КУДА ПОСЫЛАТЬ ВОПРОСЫ:
Пишите на м по адресу: **answers@linuxformat.ru** или спрашивайте на форуме: **www.linuxforum.ru**

получить файлы, Linux выдает сообщение, что не может читать папки компьютера с Vista. Могу ли я получить доступ к общему диску в Vista и вытащить оттуда файлы через Linux? Проблем с доступом к общим дискам в Windows XP, 2000 или 98 из Linux у меня не было.

JCFreak, с форума LXF

Уж сознайтесь, что Vista стоит на вашем компьютере – мы стараемся помочь всем, и не надо сваливать все на «друга»...

Лучший способ получить доступ – это использовать для монтирования дисков командную строку, тогда Вы сразу увидите, в чем причина сбоев. Сделайте это от имени суперпользователя:

```
mkdir -p /mnt/windows
mount //ИМЯ_КОМПЬЮТЕРА//C /mnt/windows -o
user=ИМЯ_ПОЛЬЗОВАТЕЛЯ
```

замените **ИМЯ_КОМПЬЮТЕРА** на сетевое имя ПК с Windows, а **ИМЯ_ПОЛЬЗОВАТЕЛЯ** на пользователя с правами администратора на данном компьютере. После этого введите пароль пользователя. Диск C будет подмонтирован (будем считать, что вы открыли доступ к нему). Не пытайтесь отключить требование пароля для сетевых дисков в Панели управления Windows: это только осложнит ситуацию. Зато вам надо включить Public Folder Sharing в секции Network And Sharing Панели управления Windows.

Но даже с этими настройками Вы не сможете зайти и скопировать любую директорию. Vista защищает директории, вложенные в пользовательские директории – вроде **ИМЯ_ПОЛЬЗОВАТЕЛЯ\PrintHood**. Однако сейчас Вы не должны иметь никаких трудностей для копирования документов и других файлов вашего друга.

Поскольку вы подмонтировали этот разделяемый диск, то можете использовать для копирования любой файловый менеджер, какой вам нравится. Вы не сказали, как хотите это сделать – соединив компьютеры напрямую кабелем или через Интернет. В любом случае все будет работать одинаково, кроме скорости передачи, но помните, что данные передаются без шифрования. Возможно, Вам придется открыть у друга порт 130 в брандмауэре или маршрутизаторе для установки соединения через Интернет. Но это позволит присоединяться кому угодно, поэтому используйте надежный пароль и закройте порт после завершения работы. По возможности, заберите компьютер друга к себе домой (или принесите к нему свой) и используйте локальное соединение по сети.

В качестве альтернативы, можно использовать программу резервного копирования Windows для создания копии данных на DVD и копирования их на компьютер с Fedora Core 6. Резервная копия Windows представляет собой zip-архив, его можно распаковать в Linux командой unzip, которая есть в Fedora Core 6. **НБ**

4 Много мониторов

Пытаюсь установить подходящий Nvidia драйвер для одно-, двух- и трехмониторной конфигурации, и после долгих поисков сделал это прямым созданием файлов конфигурации **xorg.conf** (так как GUI Nvidia постоянно жалуется на перекрытие режимов и выдает неправильную частоту обновления).

У меня теперь три разных рабочих файла **xorg.conf** – по одному для каждого необходимого режима (на один, два и три монитора), но я не могу найти информацию, как интегрировать их в одной конфигурации, чтобы переключаться между ними.

Мне нужна возможность переключаться между этими тремя режимами на лету, в идеале бы еще с использованием комбинации клавиш. А так я вручную останавливаю X-сервер, меняю **xorg.conf** и перезапускаю X-ы. Я понимаю, что нужно объединить три моих файла **xorg.conf** в один, но как? И как перезапустить X-сервер в альтернативный режим нажатием клавиш на клавиатуре (или чем-то похожим: пунктом меню, скриптом или другим способом «в один клик»)?

Я использую KDE в Fedora Core 6 и хотел бы получить указания на этот счет, но будьте великодушны ко мне – я вскочил в поезд Linux только неделю назад.

Jycle

Вы можете просто скомбинировать отдельные куски разных файлов **xorg.conf** в один, обеспечивая их разными именами. Секции Monitor могут быть расположены просто друг за другом, но вы должны быть уверены, что каждая из Ваших секций Screen имеет свое имя, с отдельными секциями для каждого из режимов. Большинство остальных записей в **xorg.conf** будут одинаковы для всех режимов – например, параметры клавиатуры, мыши, шрифтов. Затем вам необходимо создать различные секции ServerLayout для каждого режима, с различными именами – что-то вроде такого:

```
Section “ServerLayout”
  Identifier “SingleScreen”
  Screen      0 “SingleScreen” 0 0
  InputDevice “Mouse0” “CorePointer”
  InputDevice “Keyboard0” “CoreKeyboard”
EndSection
Section “ServerLayout”
  Identifier “TwinScreen”
  Screen      0 “TwinScreen” 0 0
  InputDevice “Mouse0” “CorePointer”
  InputDevice “Keyboard0” “CoreKeyboard”
EndSection
```

По умолчанию выбирается первая ServerLayout, но Вы можете указать это явно:

```
Section “ServerFlags”
  DefaultServerLayout “SingleScreen”
EndSection
```

Теперь X-сервер по умолчанию загружается в режиме одного монитора, но может быть запущен в режиме вывода на два монитора командой:

```
startx -- -layout TwinScreen
```

«--» означает «конец опций startx, передайте серверу новую информацию». Чтобы привязать переключения к клавишам, Вам пригодится маленький скрипт – сохраните его где-нибудь, например, в **/usr/local/bin/restartx**:

```
#!/bin/sh
if [[ “$(/sbin/runlevel | cut -c3)” == “5” ]]
then
  sudo /sbin/telinit 3
else
  sudo killall X
fi
sleep 2
startx -- -layout $1
```

и сделайте его запускаемым командой **chmod +x**

Коротко про...

Поиск файлов

Никогда больше не теряйте свои данные.

Linux имеет два главных инструмента для поиска файлов: *Locate* и *Find*. Первый держит базу данных о файлах вашей системы. Такой поиск быстрее, но ограничивается файлами, содержащимися в базе на момент последнего обновления. Большинство дистрибутивов устанавливают скрипт cron для поддержки актуальности базы файлов. Поиск ведется только по именам файлов. Используйте

```
locate некоторый_файл
locate –i некоторый_файл
```

(Второй вариант – для поиска, нечувствительного к регистру.)

Альтернатива – *Find*, выполняющая поиск непосредственно в файловой системе. Это гораздо медленнее, но зато информация самая свежая. Здесь также предоставляется возможность выбрать, какие директории не включать в поиск, и искать не только по имени:

```
find -name ‘*какой_то_файл*’
find /usr -iname ‘*другой_файл*’
find /usr -maxdepth 2 -iname ‘*что_за_файл*’
```

Первая строка выполняет поиск в текущей директории, заглядывая и в поддиректории. Вторая выполняет нечувствительный к регистру поиск, начиная с директории **/usr**. Третья делает то же самое, но ограничивает глубину «погружения» двумя поддиректориями.

Благодаря большому числу опций, Find является более гибким вариантом, а *Locate* великолепен для быстрого поиска. Вы можете пропустить вывод *Locate* через grep для поиска определенных директорий:

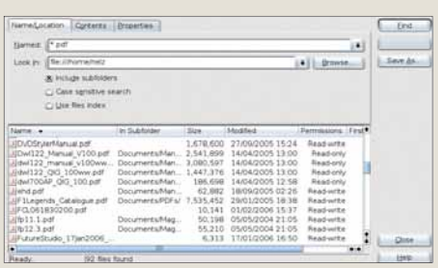
```
locate -i мой_файл | grep /home/
```

Также заметим, что *Locate* выполняет поиск в подстроках, а *Find* ищет точные совпадения.

А вы когда нибудь задумывались, где программа сохраняет свои файлы конфигурации? Выполните touch **/tmp/now**, затем запустите программу, измените настройки и выйдите. Далее наберите:

```
find ~ -newer /tmp/now
```

для поиска всех измененных файлов – сюда попадут файлы конфигурации программы и кое-какие другие.



***Kfind* – дружелюбное лицо мощной команды *Find*.**

» **/usr/local/bin/restartx**. Так как скрипт требует прав администратора для запуска, Вы еще должны отредактировать **/etc/sudoers** (также как администратор) и добавить туда следующее:

```
Ваше_имя_пользователя ALL = NOPASSWD: /usr/bin/killallX,/sbin/telinit 3
```

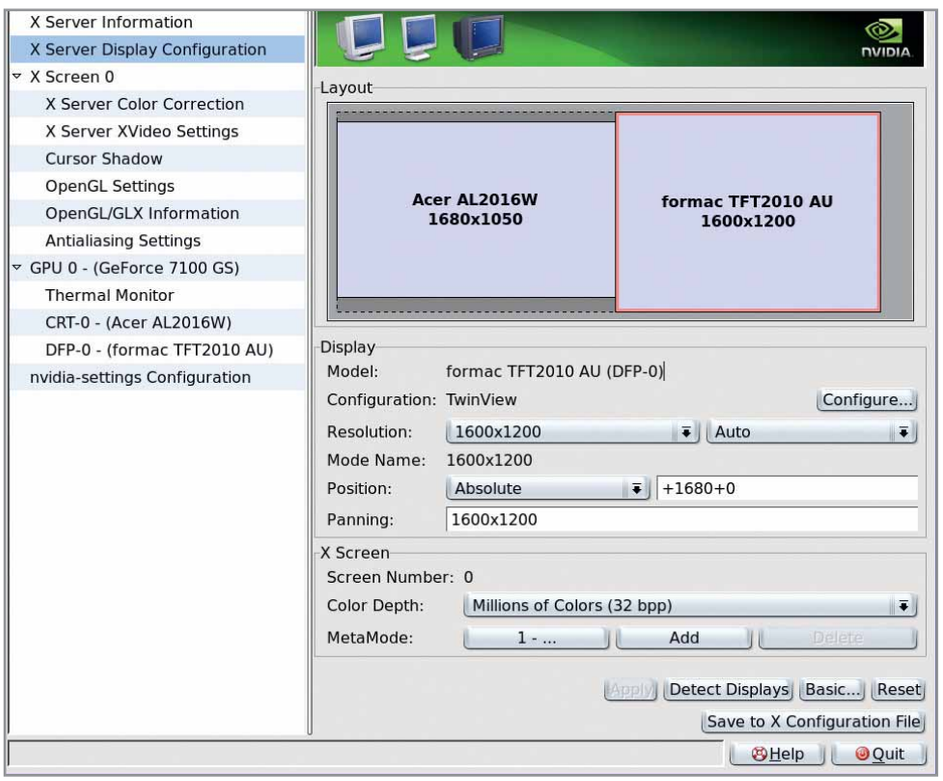
Теперь Вы сможете переключать режимы командой

```
nohup /usr/local/bin/restartx имя_нового_режима
```

nohup небходим, иначе скрипт будет уничтожен, когда закроется рабочий стол. Раз вы используете KDE, можно привязать любые команды к клавишам во вкладке Региональные и специальные возможности/комбинации клавиш Центра управления, установив по одной для включения каждого режима в Вашем файле **xorg.conf**. В конце, Вам необходимо заставить KDE «помнить» ваши открытые приложения после переключения. Чтобы сделать это, зайдите в Центр управления > Компоненты KDE > Менеджер сеансов и выберите опцию «Восстанавливать сеанс, сохраненный вручную». Это добавит вам возможность сохранять свою сессию, и Вы можете заставить скрипт делать это автоматически, дописав вторую строчку:

```
dcop ksmserver ksmserver saveCurrentSession
```

Это единственная часть, работающая только в KDE, и вы легко убедитесь, что все остальное годится и для других рабочих столов. НБ



› **Если графический интерфейс Nvidia-settings не работает, можно переключать настройки скриптом.**

5 Блокировки Samba

Я использую систему FedoraCore 1 и хотел обновить ее до Fedora Core 6. Перед этим загрузил Core 6 на отдельной машине, чтобы увидеть, как она выглядит «из коробки». Обнаружил, что sendmail настроен на доставку почты, но не смог ее принять. В Google я выяснил, что дистрибутив был выпущен с отключенной возможностью получения почты извне. Почему?

Я также настроил несколько разделяемых ресурсов в *Samba* и столкнулся со следующей проблемой: если я настраиваю директорию, скажем, **/backup**, с такими же правами доступа и владельцем, как **/var**, то я могу подключиться к нему с другой машины и обмениваться содержимым, создавать и обновлять его, а также удалять. Когда я изменил точку входа с **/backup** на **/var**, то не смог подключиться к директории. Думаю, надо изменить какие-то предустановленные параметры, но какие именно?

Чего я хочу, это открыть доступ к **/var/www/html**, чтобы «поиграть» с HTML- и PHP-файлами. Все это прекрасно работает на Core 1 и не требует изменений. Я могу перейти на Core 6 в любое время, но не сделаю этого, пока не решу эту и другие проблемы на отдельной машине.

Часто задаваемые вопросы

Брандмауэры

Убедитесь, что вы хорошо защищены от «пожаров».

» **Что такое брандмауэр, и почему я должен бояться пожара на моем компьютере?**
Брандмауэр (или межсетевой экран) представляет собой аппаратно-программный комплекс, предотвращающий несанкционированное использование сети, которую он защищает. Самым общим его применением является предотвращение попыток эксплуатации вашего компьютера или сети нехорошими обитателями Интернета. К сожалению, погасить загоревшийся компьютер он не поможет.

» **Аппаратно-программный? Так это программа или отдельное устройство?**
Бывает и то, и то. Отдельное устройство обеспечивает наилучшую защиту, поскольку оно даже не даст любым вторжениям добраться до компьютера(ов). Многие широкополосные маршрутизаторы имеют встроенный брандмауэр, так что плохие парни застрянут в вашем модеме. Другой вариант – старый компьютер с установленным брандмауэр-дистрибутивом, типа IPCop. Такая «защита» не нуждается в больших мощностях – достаточно старого 486 компьютера.

» **У меня только один компьютер. Могу я запустить брандмауэр на нем?**
Да, и достаточно эффективно. Хотя программные реализации менее надежны, чем аппаратные решения, они отлично справляются с задержанием диверсантов.

» **Какие есть варианты?**
Фактически защитой от сетевых вторжений в Linux занимается часть ядра – *iptables*. Она может быть сконфигурирована записью правил, определяющих, что и почему не разрешается. Написание правил *iptables* – сложная задача: бывает, что брандмауэр начинает работать не так, как вы ожидали, или вообще ничего не делает.

Существуют различные программы, облегчающие процесс создания правил iptables – от скриптового Shorewall (www.shorewall.net) до программ с графическим интерфейсом, вроде Guarddog (www.simonzone.com/software/guarddog) и Firewall Builder (www.fwbuilder.org). Так как все они генерируют скрипты *iptables*, вы можете использовать одну из программ с графическим интерфейсом для генерации скриптов защиты и применить их на сервере, где не запущен X.

» **Но Linux ведь безопасен, правда? Зачем мне все это?**
Это так, но только если вы следите за этим. Ведь установка замков на дверь не остановит грабителя, если окна настежь. То же самое относится и к безопасности в Интернете. Linux только предоставляет инструменты для надежной блокировки сети, а пользуетесь ими вы. Права пользователя затрудняют нарушителям некоторые вещи, но наилучший вариант – вообще никого не пускать.

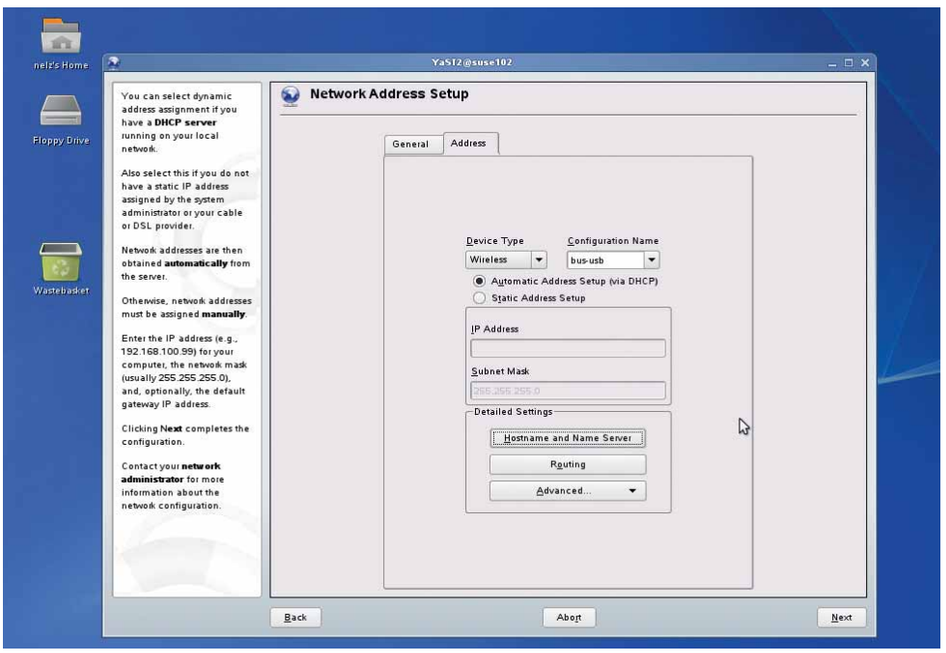
» **А у меня нечего воровать. Тогда зачем?**
Есть что, и больше, чем вы думаете. Кэш вашего браузера может содержать информацию о посещенных финансовых сайтах – хоть там и нет паролей, но вор может найти там ключ к разгадке; ваши почтовые сообщения содержат личную информацию; наконец, само ваше соединение. Им могут воспользоваться для отсылки спама, и ваш провайдер закроет вашу же учетную запись. Причем для этих пакостей не нужны права администратора, только стандартные права пользователя.

Еще один момент. Когда я модернизировал систему с Core 1 до Core 5 или 6, понадобилось несколько часов, и я счел, что проще и быстрее будет выполнить новую установку и скопировать соответствующие файлы конфигурации и данные, но теперь я не так уверен в этом.

Тони [Tony]

Похоже, что при установке Fedora Core 6 Вы выбрали режим повышенной безопасности. При этом установлен прием почты только с локальных ящиков, что легко можно изменить, и запрещено открытие доступа к «сугубо личным» директориям. Хотя после этого и возможно добиться открытия доступа к **/var**, Вам следует пересмотреть свой подход. Блокировка общего доступа к **/var** имеет понятную причину – слишком много важной информации там хранится, и очень легко сделать систему незагружаемой – по злому умыслу, некомпетентности или просто по неосторожности.

Вопрос следует задать не «как я могу поделиться содержимым **/var**?», а «нужно ли мне открывать общий доступ ко всему содержимому **/var**?» – в этом случае ответ «однозначно нет». Если Вам нужен дистанционный доступ к **/var/www/html**, то и откройте доступ к **/var/www/html**. Сделав это, Вы и снизите потенциальные риски, возникающие с открытием доступа к **/var/log** или **/var/lib**, и сможете делать то, что хотели. Есть также альтернативы использованию *Samba*. Если на обеих машинах запущен Linux, можно смонтировать **/var/www/html** на удаленной машине при помощи NFS. Если Вы используете KDE на редактируемой машине, можете избежать любой формы удаленного монтирования или открытия доступа, просто используя FISH от KDE. Этот метод использует SSH для обмена с удаленным компьютером: просто поместите fish://hostname/var/www/html в адресную строку *Konqueror* (или *Krusader*) для загрузки содержимого удаленной директории в окно файл-менеджера, где Вы сможете загрузить файлы в установленный в KDE редактор.



› Network Manager не справился с вашей беспроводной картой? Используйте стандартный метод конфигурации в Yast.

Переход с Fedora Core 1 на Fedora Core 6 – серьезный шаг. Многие ключевые компоненты будут изменены, поэтому обновление потребует гораздо больше времени, чем требуется менеджеру пакетов, когда Вы исправляете другие проблемы. Установка с нуля – наилучший вариант, но, скакнув на несколько лет вперед, Вы и обнаружили, что основные компоненты работают не так, как Вы привыкли. **MC**

6 BT без проводов

Я недавно переключился на BT Total Broadband [широкополосный канал от British Telecom в Великобритании, – прим. пер.] и пытаюсь подключить BT Home Hub, используя Wi-Fi. Установил драйвера Intel/PRO 3945abg, и *iwconfig* показывает, что сетевой интерфейс включен, но *KNetworkManager* не соединяется с хабом. Я установил шифрование на OpenSystem и ввел 40/104-битный ключ. Менеджер сети зависает после 28%, а затем переспрашивает WEP-ключ. В документации BT Home Hub говорится, что шифрование 128- »

Вопрос-победитель (русская версия)

Автор следующего вопроса, представившийся как Erlah, получает подарочный сертификат на 1000 рублей от интернет-магазина *LinuxCenter.Ru*! Просим победителя выйти на связь с редакцией: info@linuxformat.ru

Гетерогенный вопрос

Помогите, пожалуйста, решить следующую задачу. Допустим, имеется сеть на 30 машин с Windows 2003 Server в качестве контроллера домена. Необходимо настроить *Samba* на клиентской машине под управлением Ubuntu 6.10 таким образом, чтобы был прозрачный доступ к ресурсам домена без запроса пароля. Кстати, есть ли возможность зайти Windows-пользователем на Linux-машину ?

Erlah

Доступ к файловым ресурсам сети Windows-машин осуществляется с помощью *Samba*. В частности, в состав *Samba* входит утилита *smbmount*, кроме того, в Linux в виде модулей VFS реализована файловая система *SMBFS*, позволяющая выполнять монтирование файловых ресурсов SMB стандартным для Linux способом.

В качестве механизма Single Sign-On в Windows используется собственная реализация *Kerberos*. Чтобы доступ к ресурсам был прозрачен, необходимо настроить Linux-машину в качестве клиента *Kerberos* в домене (realm в терминологии *Kerberos*) Windows. Достаточно подробное описание этого процесса есть, например, здесь: http://barney.gonzaga.edu/~awithers/integration/.

После того, как Linux-клиент будет аутентифицирован в Windows-домене, можно пользоваться *smbmount* и *smbclient* с аутентификацией *Kerberos* – в этом случае пароль вводить не потребуется. Подробное описание работы с клиентом SMB есть на сайте *Samba*: http://pserver.samba.org/samba/ftp/cifs-cvs/linux-cifs-client-guide.pdf.

Пользователи Windows могут аутентифицироваться на Linux-машине со своими паролями, если в качестве PAM-модуля для соответствующего сервиса (SSH, IMAP и др.) используется **pam_krb5**, проверяющий пароли через сервис *Kerberos*. Другим вариантом является использование так называемых *Kerberized*-сервисов, в этом случае пароль вводить вообще не придется. Каждый сервис подключается к системе аутентификации *Kerberos* по-своему, например, для OpenSSH процесс описан тут: http://www.tkk.fi/cc/docs/kerberos/sso.html#openssh. Для правильной работы также требуется клиент *SSH* с поддержкой *Kerberos*, например, существует модифицированная версия *PuTTY*: http://rc.quest.com/topics/putty/.

Если речь идет о подключении пользователей Windows к файловым ресурсам Linux-машины, то это опять же реализуется средствами *Samba*, а подробное описание всего, что необходимо для работы этого программного обеспечения, находится в замечательной книге "*Using Samba*", бесплатно доступной на сайте проекта: http://us1.Samba.org/Samba/docs/using_Samba/toc.html. **ВИ**

Вопрос-победитель (английская версия)

Apache homes

Хочу настроить *Apache* так, чтобы пользователи имели персональные web-сайты в своих домашних директориях **/homes/user/website** со ссылкой на **www.blah.com/~user**. Я знаю, что могу это сделать с помощью модуля userdir. Беда, однако, в том, что пользователи монтируют свои домашние директории с Windows-машин, и когда они кладут свои файлы в папку, то *Apache* не разрешено их читать.

Как мне его настроить, чтобы файлы, положенные пользователями в их публичные папки, были доступны для чтения *Apache* автоматически? Я видел упоминание о так называемом mod rewrite, но это не ответ на вопрос. Вовсе не хочу, чтобы пользователи имели возможность менять права (слишком умно для них!) или запускать какие-нибудь скрипты каждую пару часов для проверки их прав. Есть ли модуль *Apache*, который делает что-то типа этого?

Jellyman_4eva, с форумов LXF

mod_rewrite – очень мощный инструмент, но для этой задачи использовать его неправильно, так как он переделывает переадресованные-запрошенные URL, базируясь на регулярных выражениях. А правильным выбором будет модуль userdir, который Вы упоминали. Ваша проблема сводится к обеспечению того, чтобы все HTML- и прочие файлы, которые пользователи скидывают в свое web-пространство, были читаемыми для сервера при невозможности чтения всего пользовательского каталога. Это можно легко сделать тщательным выбором владельцев и прав доступа.

При работе с модулем *Apache* userdir, **http://hostname/~username/** отображается на **/home/username/public_html/**. Первый шаг – это проверить, что директории пользователей открыты на чтение только для них самих:

```
chmod 711 /home/*
```

Затем необходимо директории **public_html** сделать доступными для чтения группе, под которой запущен *Apache*. Обычно это '*Apache*', но некоторые дистрибутивы запускают сервер под 'nobody'. Посмотрите на параметр Group в файле **httpd.conf**:

```
chgrp apache /home/*/public_html
chmod 750 /home/*/public_html
chmod g+s /home/*/public_html
```

Теперь директории пользователей доступны для чтения только для них (chmod 711), а директория **public_html** принадлежит группе '*Apache*', и члены этой группы могут ее читать (но не писать в нее). Третья команда устанавливает директорию как **setgid**, поэтому любые файлы, созданные в ней, автоматически принадлежат группе '*Apache*' вместо группы обычных пользователей. Владение же файлом остается за пользователем.

Если Вы хотите использовать другую директорию для файлов пользователя, вместо **public_html**, поправьте соответствующую часть Вашей конфигурации *Apache*. Она может варьироваться от одного дистрибутива к другому, но файл конфигурации всегда будет содержать строку:

```
UserDir public_html
```

Измените ее на ту, где Вы хотите хранить HTML-файлы в домашних каталогах пользователей. **ПХ**

» битное. Буду благодарен за любые указания, как подключить хаб.

Вот вывод *iwconfig*:

```
eth2 IEEE 802.11g ESSID:cMode:Managed
Frequency:2.412 GHz
Access Point: 00:14:7F:BE:0D:9D
Bit Rate:54 Mb/s Tx-Power:15 dBm
Retry limit:15 RTS thr:off Fragment thr: off
Encryption key:xxxx-xxxx-xx Security mode:open
Power Management:off
Link Quality=77/100 Signal level=-57 dBm
Noise level=-58 dBm
Rx invalid nwid:0 Rx invalid crypt:65 Rx
invalid frag:0
Tx excessive retries:0 Invalid misc:126
Missed beacon:0
```

Клифф Диас [Cliff Dias]

Вывод *iwconfig* выглядит правильным, кроме ключа шифрования: он слишком длинен для 64-битного, но слишком короток для 128-битного, и, вероятно, в этом и заключается проблема с шифрованием. Первым делом отключите шифрование как на Home Hub, так и на компьютере. Шифрование беспроводной передачи вещь в целом хорошая, но заставляет помучиться при настройке соединения. Гораздо проще сперва настроить незашифрованное соединение, а затем, когда соединение заработает, применить шифрование. вы можете выключить шифрование на BT Home Hub через web-интерфейс на странице администратора. Руководство скажет вам, какой адрес ввести в строке браузера, и назовет пароль по умолчанию. Кстати, вы получаете хороший шанс сменить пароль на ваш любимый.

Ваш вывод *iwconfig* показывает, что соединение работает без проблем. Проверив его работу соединением с внешней web-страницей (лучше **www.linuxformat.co.uk**, потому что Майк любит, когда накручивается счетчик посещений), можете включить WEP-шифрование снова. WEP использует так называемое 64-битное или 128-битное шифрование. «Так называемое» – потому что 24-бита из них Вам менять нельзя, отсюда и цифры 40 бит и 104 бита. 128-битный ключ вводится как 16-значная шестнадцатеричная строка, обычно разделенная дефисами для лучшей читаемости: XXXX-XXXX-XXXX-XXXX.

Если у Вас ничего не выйдет в *KNetworkManager*, попробуйте запустить *iwconfig* в терминале от имени администратора. Это может привести к выводу полезных сообщений об ошибке. Вам нужно ввести следующие команды:

```
ifconfig eth2 up iwconfig eth2 key open XXXX-XXXX-
XXXX-XXXX
iwconfig eth2 essid “BTHomeHub-8AF2”
dhcpcd eth2
```

Когда все заработает в терминале, можете поместить детали в *KNetworkManager* или отключить NetworkManager в Yast и использовать стандартный конфигуратор сети Yast.

Поиск информации в Интернет по этому поводу выдал нам больше историй проблем, чем побед. Консенсус достигнут в том, что это не особенно хороший беспроводной концентратор (несмотря на его попытки закосить под удачный аксессуар Apple) и что гораздо лучшим выбором будет беспроводная точка доступа/маршрутизатор одной из сетевых компаний. Но, учитывая, что это устройство идет бесплатно с вашим подключением, может, и стоит потратить время на его настройку. **НВ**

7 Тасуя карты

Как заставить сетевые карты помнить свои имена между перезагрузками в SUSE? У меня SUSE Enterprise 9 на моем Linux маршрутизаторе/брандмауэре, в котором установлено три сетевых карты: одна для выхода в Интернет, одна для нашей внутренней сети и еще одна для нашей DMZ [демилитаризованной зоны, – прим. пер.], они перекрывают все наши доступные извне ресурсы: web, e-mail и FTP-серверы. В основном все работает великолепно. Проблема в том, что Ethernet-имена устройств а) случайным образом меняются при перезагрузке (например, 'eth0' при следующей загрузке оказывается 'eth1') и б) любые постоянные имена, назначенные этим устройствам, вроде 'nic1' или 'nic2', часто игнорируются (хотя опция **PERSISTENT_NAME="nic1/2/3"** определена в файлах устройств **/etc/sysconfig/network/ifcfg-eth-***).

В результате при перезагрузке маршрутизатора я почти всегда вынужден запускать *ifconfig* и исправлять ID устройств в правилах *iptables* с учетом текущей (практически случайной) конфигурации устройств. Это вызывает проблемы, потому что маршрутизатор редко восстанавливается сам при любых отключениях без вмешательства.

Прилагаю файл конфигурации DMZ сетевой карты из **/etc/sysconfig/network/ifcfg-eth-id-00:02:96:00:3f:8e**. Эта карточка обычно определяется как 'eth2', и (теоретически) ей было назначено постоянное имя 'nic2' в правилах *iptables*. Когда система загружается, она иногда говорит, что устройство имеет имя 'nic2', но чаще всего этого не происходит – система просто игнорирует строку **PERSISTENT_NAME**.

› Центр управления KDE покажет версию вашего ядра, но проще воспользоваться командной строкой.

К несчастью, у меня нет других сетевых карт, чтобы попробовать что-то сделать на другой машине (с другим дистрибутивом), и я не могу остановить сервер для решения этого вопроса.

Пол Водби [Paul Waudby]

Это довольно странно – ваш файл конфигурации выглядит правильным и у нас в SUSE это работает. То, что он иногда работает – показатель, что некоторые фундаментальные куски программного обеспечения все-таки присутствуют. Не обновляли ли вы систему так, что она перешла на udev? При этом имена могут задаваться принудительно, несмотря на настройки в **/etc/sysconfig/network**. Если да, то самый простой и легкий способ решения этой проблемы заключается в установке правил именования для udev. Создайте файл **/etc/udev/rules.d/10-network.rules** от имени администратора и добавьте туда следующее:

```
SUBSYSTEM==”net”, DRIVERS==”?*”, ATTRS{address}==”00:02:96:00:3f:aa”, NAME:=”nic0”
SUBSYSTEM==”net”, DRIVERS==”?*”, ATTRS{address}==”00:02:96:00:3f:bb”, NAME:=”nic1”
SUBSYSTEM==”net”, DRIVERS==”?*”, ATTRS{address}==”00:02:96:00:3f:8e”, NAME:=”nic2”
```

заменив строки после **ATTRS{address}** MAC адресами трех сетевых карт. Хотя у SUSE имеются проблемы с повторным использованием стандартных имен, udev они не касаются. поскольку он переименовывает устройства еще до назначения любых других имен, и если вы захотите, можете здесь использовать имена **eth0/1/2**.

Вы можете обнаружить, что у вас уже есть файл в **/etc/udev/rules.d** с правилами назначения сетевых имен; тогда отредактируйте его, добавив вышеприведенные строки.

Как альтернативное решение можно использовать команду nameif для переименования интерфейсов. Это нужно делать до подключения интерфейсов. Создайте файл **/etc/mactab** для хранения списков имен интерфейсов и их MAC-адресов, например, как здесь:

```
nic0 aa:bb:cc:dd:ee:ff #internal
nic1 00:11:22:33:44:55 #external
nic2 66:77:88:99:00:aa #dmz
```

Команда **nameif** прочитает этот файл и переименует интерфейсы согласно ему. Этот метод стоит принять во внимание, если вы не используете *udev*, так как правила *udev* предлагают лучший способ для управления постоянными именами сетевых интерфейсов, и не только. **НБ**

8 Которое ядро?

Пытаюсь заставить мою систему видеть мой беспроводной USB-адаптер Belkin, используя ndiswrapper. Примечания говорят мне, что я должен иметь как минимум определенное ядро. Я новичок, так что не могли бы вы подсказать мне, где найти эту информацию? А также, не посоветуете ли, как его установить?

Барри Симпсон [Barry Simpson]

Существует множество GUI-инструментов, способных подсказать Вам, какое ядро Вы используете: Центр управления KDE показывает версию на стартовой странице, или Вы можете использовать менеджер пакетов дистрибутива для нахождения версии пакетов с ядром (некоторые дистрибутивы называют их ‘linux’). Самый же легкий путь – это открыть терминал и набрать одну из команд:

```
uname --kernel-release
uname -r
```

До ndiswrapper дело может и не дойти: некоторые беспроводные устройства Belkin имеют родную поддержку. В этом случае запустите

```
sudo lsusb
```

в терминале, чтобы узнать больше о вашем устройстве. Затем поищите в Google или на форуме вашего дистрибутива [российские пользователи могу поискать на **www.linuxforum.ru**, – прим. пер.] информацию об этом устройстве. Вы также можете найти информацию о наилучшем драйвере для вашего устройства на **http://qbik.ch/usb/devices**. Если же родного драйвера нет, то без ndiswrapper не обойтись. При этом важно использовать драйвер, прилагавшийся к устройству: производители имеют привычку менять внутреннюю начинку устройства, оставляя один тот же номер модели, поэтому драйвер на вид идентичного устройства может оказаться бесполезным. Если в вашем дистрибутиве (вы не упомянули, какой он у вас) есть инструмент для настройки беспроводных устройств, лучше используйте его вместо ручной настройки. Некоторые из них, например, Yast от SUSE, также могут настроить *ndiswrapper*. **MC**

9 Любопытной Варваре...

У меня есть небольшая локальная сеть на 500 компьютеров с интернет-шлюзом на базе Linux (Ubuntu 6.06). На шлюзе работает биллинговая система и Jabber-сервер. В связи с тем, что пользователей достаточное количество, всегда находится какой-нибудь «умник», который страсть как желает перехватить передаваемые данные (пароли, файлы, личную переписку) каким-нибудь сниффером со своего рабочего места, а после использовать их в своих целях. Вопрос в следующем – каким образом можно отлавливать подобных личностей и как обезопасить сеть от перехвата данных на программном уровне?

Денис Суханов

Защита от перехвата данных в Jabber (правильнее сказать – в XMPP), как и в других протоколах реализована стандартным способом – шифрованием (обычно SSL/TLS). Оно поддерживается большинством популярных клиентов (по крайней мере, Gajim и PSI), и с этой стороны не должно быть никаких проблем. Если же используемые вами протоколы не поддерживают шифрования напрямую, может пригодиться *stunnel* (**www.stunnel.org**). Его настройка была подробно рассмотрена Андреем Бешковым в статьях для журнала «Системный администратор», которые сейчас можно найти в сети: **http://onix.opennet.ru/content/view/34/26/**.

Поиск работающих снифферов – более сложная задача, не имеющая 100%-но надежного решения. Один из методов использует тот факт, что сетевая карта злоумышленника переводится в беспорядочный режим (promiscuous mode), т.е. отлавливает все пакеты, идущие по сети. Заполнив ее кадрами для несуществующих MAC-адресов, можно вызвать перегрузку на машинах, где запущен сниффер (остальные узлы не будут обращать внимания на эти пакеты на аппаратном уровне, а снифферу придется декодировать и визуализировать их). Повышенная нагрузка скажется на среднем времени ICMP-ответа («пинга»). Я бы не рекомендовал писать служебную записку на основании такого анализа, но если какой-то компьютер начинает задыхаться под нагрузкой пакетов для несуществующих адресов, это лишний повод заглянуть к его хозяину и посмотреть, в чем дело. **BC**

10 Puppy по-русски

Меня заинтересовал дистрибутив Puppy Linux, о котором вы писали в **LXF93**. Я испробовал все возможности, о которых указал в своей статье Сергей Супрунов, однако, мне не удалось настроить использование русского языка в тестовом редакторе *Abiword* и электронных таблицах. Хотелось бы узнать есть ли поддержка русского языка и как ее настроить. Сейчас я работаю в Windows, но хочу побольше узнать о Linux и со временем думаю перейти на него, а *Puppy* мне понравился тем, что хорошо работает на старых машинах. Сейчас у меня опыта работы в Linux практически нет.

Иван Герасименко

Если я Вас правильно понял и под использованием русского языка подразумевается возможность вводить символы кириллицы, то вот инструкция по настройке русской раскладки:

1. Щелкните на иконке Setup или выберите пункт Menu > Setup > Wizard Wizard.
2. Перейдите в Change the mouse and keyboard.
3. Щелкните Advanced Xorg keyboard configuration.
4. В открывшемся окне XKB_Config_Main следует:
   - задать модель клавиатуры – Keyboard Model (при желании)
   - щелкнуть Layouts и задать основную раскладку (ru)
   - щелкнуть Layouts еще раз, указать, что хотите добавить (Add) раскладку, выбрать us
   - щелкнуть Layouts еще раз, указать, что хотите добавить (Add) раскладку, выбрать ru (с первого раза она почему-то не сохраняется, по крайней мере, в моей инсталляции)
   - щелкнуть Layout Variants и выбрать ru: Winkeys (вы должны видеть варианты как для us, так и для ru – иначе нужно вернуться в Layouts и в очередной раз добавить недостающую раскладку)
   - щелкнуть внизу кнопку Yes, чтобы изменения вступили в силу.
   - щелкнуть Group Shift/Lock behavior и выбрать желаемую комбинацию клавиш для переключения раскладок (мне больше по душе grp:alt_shift_toggle)
   - при желании здесь же можно настроить и другие интересные параметры
   - щелкнуть внизу кнопку Yes, чтобы изменения вступили в силу.

После этого никаких проблем с раскладками наблюдаться не должно. **CC**

Мы стараемся ответить на все вопросы. Если вы не нашли ответ на свой, это, возможно, потому, что мы уже ответили на похожий вопрос.

Большой вопрос

На этом раздел заканчивается, но наши эксперты не расходятся по домам, пока не дойдут до сути общей проблемы

11 Взрослая проблема

У моего сына PlayStationPortable. Я хочу конвертировать DVD и другое видео в MPEG4, чтобы он мог смотреть их в долгих путешествиях. Уверен, что *Transcode* или *Mencoder* на это способны, но их man-страницы – сплошной жаргон. Есть ли какой-нибудь простой способ конвертации видео для PSP?

Льюис Берфорд [Lewis Burford]

О Да, есть! Когда конвертируешь DVD, наипростейшей программой является *dvd::rip* – графическая оболочка для *Transcode*, *MPlayer* и других. К сожалению, в ней нет поддержки варианта MPEG4, используемого в PSP, поэтому Вам понадобится *FFmpeg*, другая консольная программа, но без запутанных опций *Transcode* или *Mencoder*. Графический интерфейс к *FFmpeg* называется *Vive*, его можно найти по адресу http://vive.sourceforge.net.

Он существует только в виде исходных кодов, но очень легко устанавливается, при наличии установленных инструментов для компиляции. Скачайте последний tar-архив с сайта (сейчас это 2.0.0-beta1) и установите его, набрав в терминале

```
tar xf vive-2.0.0-beta1.tar.gz
cd vive-2.0.0-beta1
./configure
make
su -c "make install"
```

При запросе, введите пароль администратора. Пользователи Ubuntu должны заменить последнюю команду на нижеследующую и использовать свой собственный пароль:

```
sudo make install
```

Vive теперь можно найти в меню KDE или Gnome или запустить его, набрав в командной строке **vive**. *Vive* использует предустановки для хранения настроек типов вывода. Пример файла конфигурации по умолчанию не ставится; сделайте это вручную, выполнив

```
mkdir ~/.vive
cp /usr/share/doc/vive/examples/preferences
~/.vive
```

Этот файл уже содержит предустановки видео для iPod/PSP, но не позволяет генерировать PSP-совместимые файлы и не справляется с широкоформатным видео.

Добавьте следующие строки к файлу настроек:

```
[PSP]
format=psp
vcodec=mpeg4
maxrate=768000
bitrate=700000
bufsize=4096
aspect=4:3
width=320
height=240
acodec=aac
ab=64
ar=24000
comment=Encoded by Vive
```

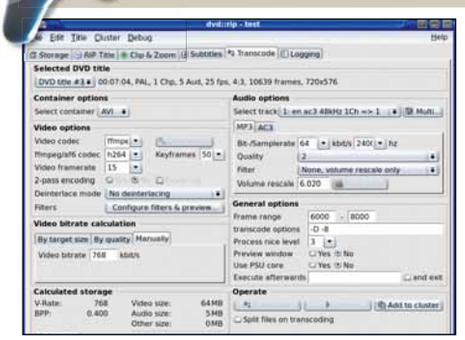
Для широкоформатного видео скопируйте блок, переназовите его, к примеру, PSPWide и измените значения aspect, width и height на 16:9, 368 и 208 соответственно.

Запустив *Vive*, Вы должны выбрать или главу DVD, или файл для кодирования – нажмите Load, и *Vive* прочитает заголовки с DVD. Затем выберите выходной файл и используемые установки. Вы также можете изменить значения аудио и видео декодирования по умолчанию, выбрав другие пресеты. Видеофайлы должны быть записаны в директорию **/MP_ROOT/100MNV01** на карту памяти и названы **M4V00001.MP4**, **M4V00002.MP4** и так далее.

За один раз *Vive* может конвертировать только один файл, но программу можно запустить из командной строки в режиме пакетной обработки. Для конвертации всех AVI файлов из директории используйте следующий скрипт:

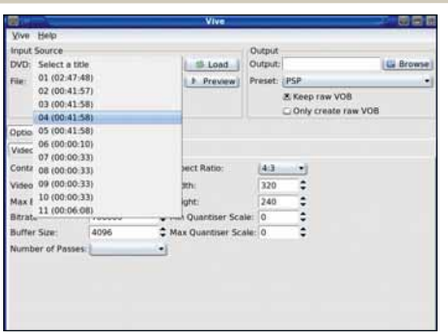
```
for FILE in *.avi
do
vive -p PSP -i $FILE -o ${FILE/.avi/.mp4}
```

Шаг за шагом Конвертируем видео для PSP



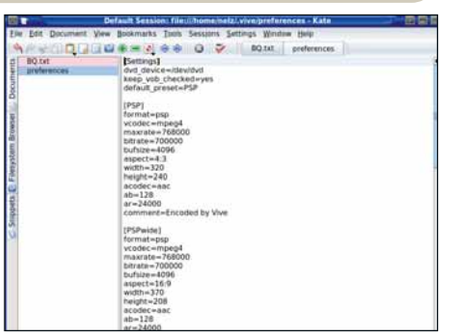
1 Не сейчас

dvd::rip хорош для декодирования видео и DVD, но не может создавать файлы PSP... пока.



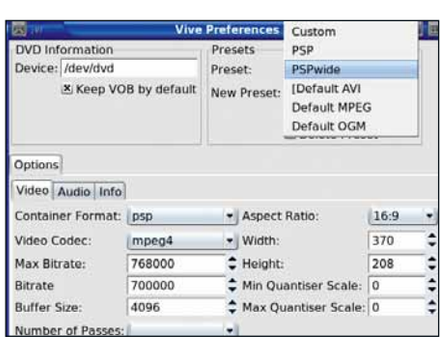
2 Выберите дорожку

Выберите главу вашего DVD, или выберите файл для декодирования. За раз *Vive* справляется только с одним.



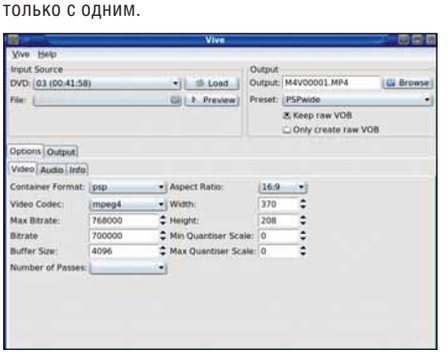
3 Задайте установки

По умолчанию настройки *Vive* содержат установки для iPod/PSP. Но тут создаются два пресета для PSP.



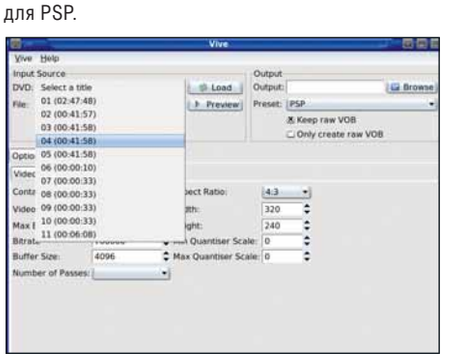
4 Выберите предустановку

Здесь можно выбрать предустановку для PSP и изменить настройки по умолчанию до начала кодирования.



5 Дайте файлу имя

PSP привередлива к именам файлов – даже из-за лишнего 0 в номере файл станет невидим.



6 Любуйтесь работой

Вкладка Output показывает… угадали, вывод, но в ней можно также запускать команды.

Технология счастья

SUNRADIO.RU

сетевое радио под ключ на базе Linux

новое будущее вашей компании

pr@sunradio.ru +7 812 955 76 70 www.sunradio.ru

LINUX FORMAT

2 года с вами

24 шага на пути к свободе

присоединяйтесь!

LXF DVD96

Не с одинарной, не с двойной, а с тройной загрузкой!

Майк Сондерс любовно подбирает содержимое диска Linux Format, а также поддерживает сайт www.linuxformat.co.uk.

Переосмысляя рабочий стол

Оглянемся на пару десятилетий: многое ли изменилось в пользовательском интерфейсе? Помню, как мы в конце 80-х использовали Amiga Workbench, восхищаясь суперскоростным интерфейсом под управлением мыши и предвкушая грядущие чудеса. Apple вместе с Mac'ом принесла в массы графический пользовательский интерфейс (GUI), Microsoft пристегивал собственную оболочку к DOS, и даже у старой развалины C64 был собственный GUI в форме GEOS. Таким виделось будущее – абсолютно у всех в компьютерном мире текли слюнки в ожидании интерфейсов с невероятным дизайном, которые обязательно должны были появиться.

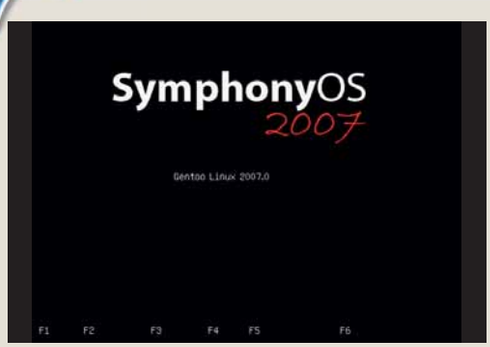
И вот, 20 лет спустя, наши рабочие столы выполнены практически в той же самой манере, что и у старых добрых Mac OS и Windows 1.0. Разумеется, у нас есть такие классные эффекты, как Aero, Aqua и Compiz, и концепция рабочего стола претерпела множество мелких доработок – например, переключение окон посредством Alt+Tab и OS X Exposé. Но перед нами все та же фурнитура в виде значков рабочего стола, панелей меню и т.п., и по-прежнему сохраняется идея, что наши рабочие столы – это файлы внутри папок.

Кто-то скажет, что верность устоявшимся метафорам рабочего стола – это хорошо. Однако мы в итоге потерялись среди файлов и папок, минимизируя окна, чтобы подобрать недоделанный документ, разместившийся на рабочем столе. Может, есть и другие пути. Возможно, привычный рабочий стол не слишком эффективен, и стоит взглянуть на альтернативные варианты? Именно этим занимается Symphony OS – полная информация на стр. 98.

На DVD этого месяца есть еще масса интересного для исследования, включая фаворита компьютерных гуру – Gentoo, пререлиз KDE 4 и множество новых программ. Наслаждайтесь, а если у вас есть комментарии или предложения – обязательно свяжитесь с нами!

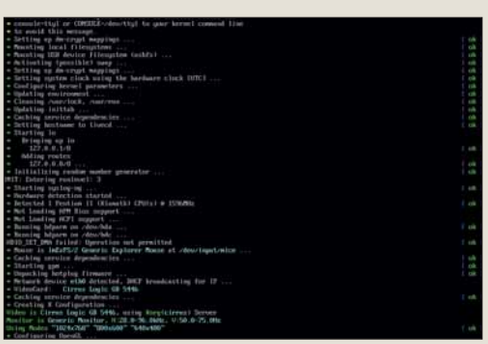
mike.saunders@futurenet.co.uk

Шаг за шагом: Установка Gentoo 2007



1 Загрузка

Загрузите компьютер с **LXF DVD** и нажмите Enter, чтобы выбрать верхний раздел в меню загрузки. Если возникнут проблемы с графикой, попробуйте безопасный видеорежим safe video.

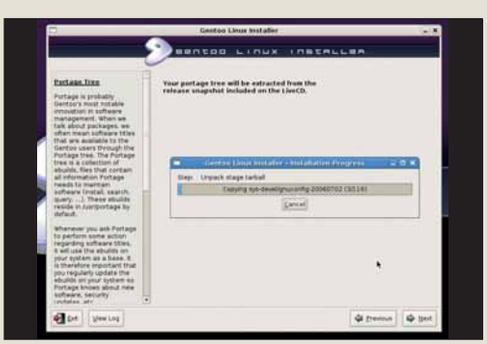


2 Запуск

Gentoo загрузится прямо с диска, обеспечивая вас обратной связью в течение всего процесса. Внимательно следите за сообщениями об ошибках.

5 Деление диска на разделы

Выделите не менее 5 ГБ под корневой раздел (/), **ext3**, и не менее 512 MB под раздел подкачки [swap]. Чтобы получить более подробную информацию, читайте подсказку слева.



6 Portage

После настройки необходимых сетевых соединений программа установки Gentoo начнет копирование системы *Portage*.

Дистрибутив Linux

Gentoo 2007.0

За время своего существования Gentoo претерпел несколько трансформаций имиджа. Появившись в качестве дистрибутива с управлением пакетами в стиле BSD, он быстро обрел высокую репутацию «дистрибутива лихачей» – ОС для тех, кто помешан на скорости и тонкой настройке. Однако вскоре это все отмерло, и сегодня Gentoo уважаем как мощная платформа для желающих создать собственный отличный дистрибутив, а не довольствоваться настройками по умолчанию. Тут не обойтись без хорошего знания технологий Linux, но оно позволит вам создать машину на ваш вкус.

В большей мере это происходит благодаря системе *Portage*. Gentoo выбирает основанную на исходниках систему установки программ, позволяющей так отбирать библиотеки и оптимизации, нужные вам в пакете, чтобы не пришлось вместе с ними получать тонны ненужного мусора. Плюс к тому, вы можете обновлять программы тогда и так, как вы этого желаете. Релиз Gentoo 2007.0 содержит два значительных усовершенствования:

» Новая программа установки Программа установки Gentoo была полностью переписана, при этом исправлены основные ошибки предыдущих версий, а ее запуск и работа стали проще.

» Обновления системы Большинство пользователей Gentoo делают установку один раз, а затем постоянно обновляют ее, но для новообращенных в этом релизе имеется ядро 2.6.19, GCC 4.1, *glibc 2.5*, KDE 3.5.5, Gnome 2.16.2, *OpenOffice.org 2.1.0* и *Firefox 2.0.0.3*.

Версия на нашем DVD, запускающаяся непосредственно с диска – Live-дистрибутив, позволяющий загружать и использовать Gentoo 2007.0 без установки на жесткий диск. Это превосходный способ познакомиться с Linux: просто загрузите его с DVD и нажмите reset по окончании знакомства. Можете продемонстрировать Linux своим друзьям и сотрудникам, не боясь угробить установленные программы Windows.

Если вы созрели для установки Gentoo – он знаменит своей превосходной документацией, включающей очень подробный документ на рабочем столе, который объясняет весь процесс. Он просто всеобъемлющ, и может даже выбить вас из колеи, если вы – новичок в ОС; однако если вы какое-то время уже работаете с Linux, вы оцените его дотошность.

Сообщество Gentoo создало отличную индивидуально настраиваемую систему самопомощи, и высоко ценит всех, кто вышел за рамки привычных решений и предварительно настроенных пакетов. Gentoo идеален для тех, кто хочет полностью контролировать свою систему. Если вы готовы к исследованиям нового, загрузитесь с нашего диска, следуйте изложенным ниже инструкциям по быстрому запуску и документации на рабочем столе. А если вы споткнетесь, дружественные форумы на http://forums.gentoo.org очень быстро поставят вас на ноги.

» Live-дистрибутив обладает отличным рабочим столом, на котором и Gnome, и *Firefox*, и *OpenOffice.org* – все под рукой.

3 Установка

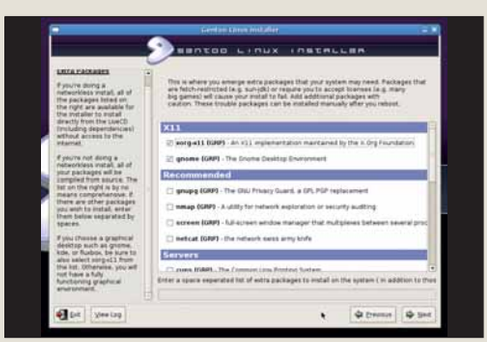
Когда появится рабочий стол, дважды щелкните по GTK+ значку Программы установки Gentoo Linux, чтобы начать процесс инсталляции.

4 Режимы

Для быстрой установки, не требующей сетевого соединения, выберите Networkless («Без сети»). Информацию по остальным режимам см. в полном руководстве на рабочем столе.

7 USE-флаги

Вы можете настроить USE-флаги – это опции сборки именно под вашу машину. Затем задайте пароль для root, и начнется копирование ядра.



8 Программы

Создайте учетную запись обычного пользователя и выберите нужные вам пакеты. Они будут установлены, вы произведете окончательную настройку, после чего можете перегрузиться и войти в свежеустановленную систему Gentoo!

Дистрибутив Linux

Symphony OS 2007

Symphony OS пару лет пробуксовывала на одном месте, угодив в раздел «эзотерических» дистрибутивов на web-сайтах, посвященных Linux, из-за своей капитально непробиваемой среды рабочего стола. Но после некоего периода ожиданий разработчики переключились на базу Ubuntu, и хотя многое еще предстоит сделать, система уже заслуживает ознакомления.

Сначала небольшая предыстория. Актер дубляжа Джейсон Спайсек [Jason Spisak] (www.imdb.com/name/nm1032853), долгие годы страдавший «из-за безмозглых интерфейсов, созданных профессионалами», решился создать собственное руководство по GUI. Он подытожил свой опыт работы в разных версиях Windows, Mac OS и Linux и разработал список правил, которым должны подчиняться пользовательские интерфейсы. Но он не только теоретизировал, а и начал создавать основанную на *FVWM* среду рабочего стола Mezzo, чтобы воплотить теорию в жизнь.

Главная цель Mezzo, как можно видеть в Symphony OS – отказ от представления, что рабочий стол – это папка. В Mezzo нет рабочего стола как такового: вместо него там присутствуют предопределенные «пространства» для запуска программ или открытия файлов. Также там присутствуют «десклеты» [desklets] – небольшие программки, предоставляющие информацию: например, о погоде или о биржевых котировках.

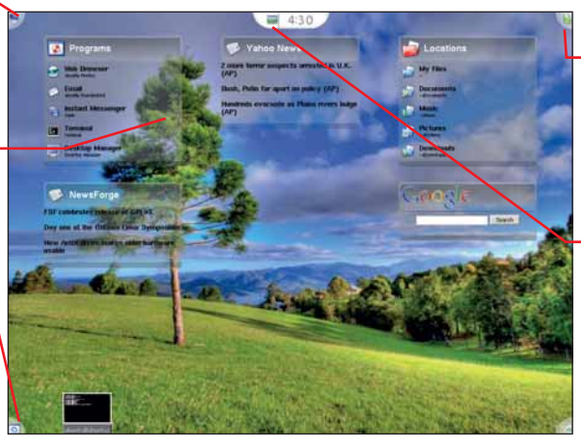
› Рудиментарные инструменты настройки Mezzo требуют доработки.

Следующей основной функцией Mezzo является прочная привязка к закону эргономики Фиттса [Fitts' law], гласящего, что интерфейсы должны использовать границы экрана. Поэтому в Mezzo углы экрана используются для переключения на разные типы рабочих столов, то есть вы можете просто подвести мышь к углу экрана, вместо тщетных попыток попасть в крошечную кнопку. Спайсек отказался также и от кучи параметров настройки и вызывающих раздражение всплывающих диалоговых окон.

Symphony OS – это основанный на Ubuntu «живой» дистрибутив, который загружает Mezzo, и поскольку он не трогает жесткий диск, можете изучать новый рабочий стол и не беспокоиться ни о каких последствиях. Дистрибутив содержит такие популярные программы, как *Firefox* и *Thunderbird*, однако не расслабляйтесь – будут ошибки, потому что он все еще находится в стадии разработки. И все же он чарует исследователя, поэтому загрузите его с нашего диска (32-битовый, но должен работать и на 64-битовых системах) и попробуйте.

Помните, что вы можете установить Symphony OS на жесткий диск, открыв терминал и введя *sudo ubiquity* (значок не работает). Однако мы не советуем этого делать, если вы пока что не ас Linux. Дистрибутив требует не менее 256 МБ ОЗУ, и пароля для пользователя по умолчанию нет (поэтому *sudo bash* запускает оболочку `root`). Заходите на www.symphonyos.com – там вы найдете последние новости, и на наши форумы на www.linuxformat.co.uk/forums – если понадобится помощь.

Знакомство с рабочим столом Symphony OS



Компьютер
Щелкните по значку в верхнем левом углу для просмотра устройств, задач и настройки.

Панели
Эта панель обеспечивает доступ к вашим любимым приложениям – нажмите на нее, чтобы запустить их.

Программы
Щелкните по значку внизу слева – и перед вами появится подробный список программ.

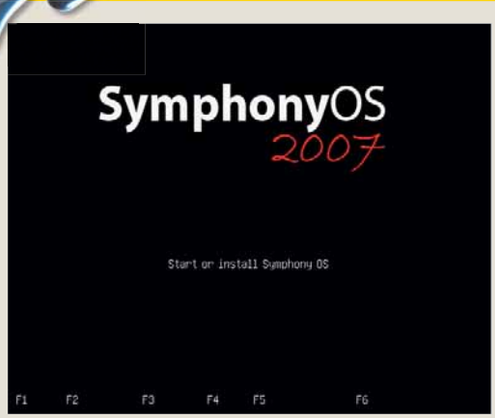
Файлы
Щелчок по этому значку активирует (слегка нестабильный) файловый обозреватель.

Минимизировать
Щелкните по маленькому значку рядом с часами, чтобы минимизировать все открытые окна.

Вход в систему

Войдите как пользователь 'linux' с паролем 'linux'. Пароль администратора **root** – тоже 'linux'.

Шаг за шагом: Запускаем Symphony OS 2007



1 Запуск
Запустите компьютер с **LXF** DVD в дисководе, с помощью стрелок выберите Symphony OS, затем нажмите Enter (если возникнут проблемы с изображением, попробуйте опцию safe graphics).

2 Загрузка
Symphony OS загрузится непосредственно с DVD – она и не притронется к вашему жесткому диску. Это может занять пару минут, в зависимости от скорости вашего компьютера.

Живой дистрибутив

KDE 4 Beta Live

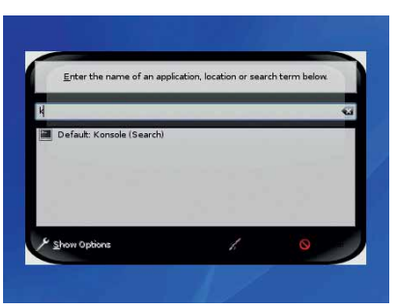
Мир Linux просто гудит от переполняющих его новостей и сплетен о KDE 4. Самый популярный рабочий стол переживает весьма основательные изменения и перепланировку. С момента последнего релиза KDE прошло немало времени (версия 3.0 появилась в 2002 году), поэтому версия 4.0 обрела почти что мифические качества. Процесс разработки не был гладким: предлагались, рассматривались и исключались новые инфраструктуры, но команда разработчиков подбирается все ближе к цели, и есть надежда, что окончательный релиз KDE 4.0 появится в октябре.

Однако, из-за чего столько шума и волнений? KDE 4 будет первым релизом, использующим *Qt 4*, который, по расчетам разработчиков, быстрее примерно на 30 процентов. Новая подсистема мультимедиа, *Phonon*, должна предоставить программистам богатый выбор инструментов для работы с аудио- и видеофайлами, позволяя им цепляться за такие матерые движки, как *GStreamer* и *Xine*.

Новый компонент *Solid* имеет целью упростить работу с устройствами, иными словами, вы подключаете что-нибудь – скажем, цифровую камеру – и программы тут же могут начинать с ней работать. Немало было сделано и в области пользовательского интерфейса: проект Oxygen разработал для KDE новые значки и графику, стремясь обеспечить единый внешний вид всех приложений и поддержку современных стандартов именования.

Еще есть *Plasma*, которая соединяет рабочий стол, панели и виджеты (*SuperKaramba*) в одну программу, создавая идеальный новый и свежий GUI, который может потягаться с Mac OS X.

На данный момент KDE 4 все еще находится в стадии интенсивной разработки, когда все время что-то выходит из строя и подвергается изменениям. Но мы рады предоставить вам первую бета-версию, чтобы вы смогли познакомиться с новыми возможностями, используя все удобства «живого» дистрибутива. Вам не придется ничего устанавливать, просто загрузите компьютер с нашего диска, выберите KDE 4 Live из меню, и – вперед. Если вас интересует завтрашний день графических рабочих сред в Linux, вы не должны это пропустить. Чтобы запустить KDE 4 Beta 1 Live, вам потребуется не менее 512 МБ ОЗУ. Используйте ‘linux’ в качестве имени пользователя и пароля. (пароль root, если вдруг понадобится, тоже ‘linux’). Мы не станем описывать рабочий стол – вам будет интереснее познакомиться с ним самостоятельно – но укажем несколько интересных направлений. Попробуйте щелкнуть по значку папки Home на панели, чтобы увидеть *Dolphin* – новый, упрощенный файловый менеджер KDE 4, который заменил собой *Konqueror*. Последний по-прежнему присутствует в качестве web-браузера, и теперь у него значительно измененный (и более ясный) интерфейс. Если на рабочем столе отсутствуют значки, запустите **kdesktop** через K-меню (диалог **Run Command**). Вы также можете помочь разработчикам, присоединившись к ним на www.kde.org.



› Даже диалоговое окно Run Command обзавелось новой графикой.

› *Dolphin* – новый менеджер файлов KDE 4, предпочитающий простоту богатому выбору функций Konqueror.

3 Знакомство

Знакомьтесь с рабочим столом, как было описано выше; по окончании щелкните по значку сверху-слева и выключите компьютер на панели справа. После этого можете извлечь диск.

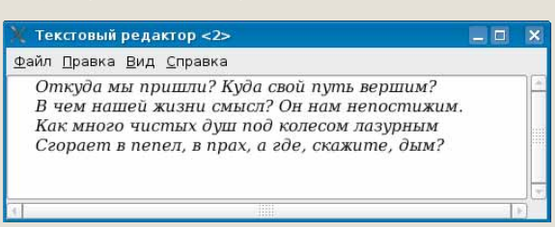
Журнал

Программирование на GTK+

В этом месяце мы рады предложить вашему вниманию очередную порцию отечественных статей – полный набор уроков *GTK+* **Андрея Боровского**, публиковавшихся в нашем журнале с декабря 2006 по август 2007 года. Вы найдете их в разделе **Журнал/PDFs**; для просмотра можете использовать *Evince* в Gnome или *KPDF* в KDE.

Мы начнем с азов, и наша первая программа не будет иметь иных функций, кроме выхода (что в терминах *GTK+* само по себе является не таким уж тривиальным действием). На следующем уроке обсуждаются вопросы интернационализации (невниманием к ним грешат многие зарубежные учебники) и менеджеры компоновки, после чего мы переходим к системе сигналов, используемой *GTK+* – это тот механизм, который обеспечивает реакцию виджетов на действия пользователя и другие события. После этого мы изучим визуальное проектирование интерфейсов – как средствами *Glade 2.x* (т.е. с генерацией исходного кода на C), так и в стиле *Glade 3.x* (с динамической загрузкой XML-описания интерфейса).

К шестому уроку вы узнаете о *GTK+* достаточно, чтобы перейти к рассмотрению нового набора API, специфичного для Gnome. Однако процесс ознакомления с *GTK+* на этом не заканчивается: попутно с изучением Gnome VFS вы узнаете о *GtkTextView* – мощном виджете, позволяющем создать полноценный текстовый процессор а-ля *OOo Writer*, и научитесь обращаться с буфером обмена. А завершит эту серию учебников разговор о *gtkmm* – привязке *GTK+* для языка C++.



› Каждый программист в своей жизни должен написать текстовый редактор.

Прочее ПО

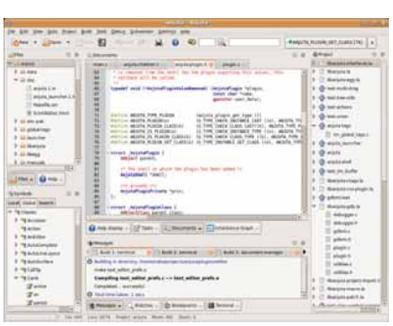
Новые релизы

Как обычно, на нашем диске полным-полно новых релизов. В разделе **Рабочий стол** вас ждет *Gimmage*, быстрая и нетребовательная к памяти программа просмотра изображений, которая вписывается в Gnome или Xfce. Она не претендует на звание полновесного редактора, но имеет все основные функции, вроде увеличения размера изображения и поворота. Вы найдете на диске исходный код, а также deb-пакет для систем Ubuntu и Debian.

Взгляните также на *gjots*, систему для заметок с тузом в рукаве: вместо того, чтобы записывать напоминания в виде простого текста, *gjots* дает вам возможность разместить информацию в виде древовидной структуры. Возможно, у вас, как и у нас, есть привычка скидывать обрывки информации (списки дел, адреса, заметки и т.п.) в один-единственный файл. *gjots* помогает выстроить информацию в иерархическом порядке, значительно упрощая ее восприятие. На диске вы найдете исходные тексты и RPM-пакет, который можно превратить в deb утилитой *alien*.

> «Nero бросает вызов K3b за корону «короля прожига» под Linux.»

В разделе **Рабочий стол** представлена демо-версия *Nero Linux* – программы для записи CD и DVD, с которой мы уже встречались на стр. 16. Nero соперничает с *K3b* с явным намерением отнять у последнего корону «короля прожига» в Linux, и хотя данная версия не так функциональна, как ее Windows-аналог, она уверенно развивается и уже предлагает поддержку Blu-ray и HD DVD. *Nero Linux* можно установить в виде RPM (для Fedora, Mandriva и SUSE) или deb-пакета (в Ubuntu или Debian), доступны 32- и 64-битные сборки. Демо-версия будет работать в течение недели – если захотите пользоваться этой программой и далее, вам придется приобрести серийный ключ на www.nero.com.



› ***Anjuta*** **удерживает позиции одной из лучших IDE Linux в своем новом релизе 2.2.**

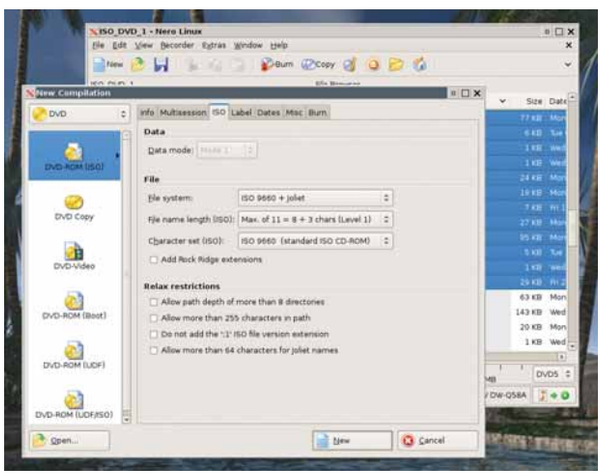
Кодирование и сеть

Если ваши пальцы кодера зудят от желания испробовать новую IDE, добро пожаловать в наш раздел **Разработка**, где вы найдете новехонький релиз *Anjuta 2.2*. Эта многоуважаемая среда постоянно отодвигает пределы совершенства: теперь в ней имеется автодополнение в компоненте редактора, новый шаблон SDL в Мастере Проектов (Project Wizard), множество обновлений и исправленных ошибок. Для *Anjuta* нужны *GDL* и *Gnomebuild*, которые мы разместили в папке **dependencies**.

Еще один новый релиз – *Vim 7.1*, обновление для текстового редактора, который вызывает или пылкую любовь, или лютую ненависть. Да, привязка клавиш *Vim* может показаться какой-то инородной пыткой тем, кто знаком с «нормальными» текстовыми редакторами, однако нельзя отрицать, что это один из самых мощных существующих редакторов, и у него множество поклонников. В *Vim 7.1* нет каких-либо потрясающих новых функций, но исправлено множество ошибок и усовершенствованы файлы поддержки проверки синтаксиса.

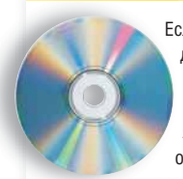
В разделе **Интернет** много всяких полезных штук, в том числе *KTrafficAnalyzer* –утилита KDE, отображающая, как вы пользуетесь сетью. Вы сможете просматривать журнал вашей работы в сети, чтобы отслеживать периоды наивысшей активности – это бонус, если у вас есть ограничения по трафику и вы хотите быть уверены в том, что не использовали больше положенных вам гигов. Также в этом разделе имеется *Empathy*, молодой, но быстро растущий клиент обмена сообщениями, вместе с *GAdminHTTPD*, обеспечивающим дружественный административный интерфейс для web-сервера *Apache*.

А в разделе **Офис** вас ждут обновления для двух основных офисных пакетов с открытым кодом, *OpenOffice.org* и *KOffice*. Удалите все ранее установленные версии *OpenOffice.org*, распакуйте файл **.tar.gz** и установите содержащиеся в нем RPM (или конвертируйте их в deb, если у вас установлен Ubuntu или Debian). *OOo* установится в **/opt** – запустите *programs/soffice* или создайте новую ссылку в меню для запуска офисного пакета. В свою очередь, *KOffice 1.6.3* можно установить или из исходных текстов (при условии, что у вас наличествуют пакеты разработки KDE и *Qt*) или из deb/RPM, если вы работаете с Ubuntu 7.04 или OpenSUSE 10.2. Для Ubuntu, надо запустить **dpkg -i *.deb** (от имени администратора) после деинсталляции всех ранее установленных версий *KOffice*; для OpenSUSE, произведите установку с помощью *YaST* или введите в соответствующей директории **rpm -Uvh *.rpm**.



› **Демо-версия из раздела Рабочий стол позволит вам неделю резвиться с самым свежим предложением от *Nero*.**

И напоследок...



Если вы вволю надистрибутивились, перейдите в раздел **Игры** с развлечениями для Linux. Гвоздь сезона в этом месяце – *Defcon*, сюрреалистический военный симулятор, основанный на MAD (Mutually Assured Destruction – взаиморазрушение). Полный обзор *Defcon* вы найдете на стр. 12 – на нашем диске представлена демо-версия, которая ограничивает ваши возможности игрой против одного соперника, а не шести – это может быть компьютер или человек. Продвинутые режимы тоже отсутствуют. Извлеките **defcon-v1.42.tar.gz** в домашнюю директорию, и, чтобы начать игру, запустите **defcon** из полученного каталога.

Фанаты игр в стиле ретро должны познакомиться с *Epiphany* (не путать с web-браузером Gnome), клоном игры всех времен и народов *Boulderdash*. Это не самая зрелищная игра на планете, но она соединяет в себе все классические элементы. И, наконец, кто интересуется байкерскими трюками в стиле *Kickstart*, взгляните на *X-Moto*. LXF

› **Два колеса – хорошо, четыре – плохо! Байкерский беспредел – это в порядке вещей в *X-Moto*.**

ВНИМАТЕЛЬНО ПРОЧТИТЕ ЭТО ПЕРЕД ТЕМ, КАК ИСПОЛЬЗОВАТЬ DVD-ДИСК !

Информация о диске

ЧТО-ТО ПОТЕРЯЛИ?

Часто случается, что новые программы зависят от других программных продуктов, которые могут не входить в текущую версию вашего дистрибутива Linux.

Мы стараемся предоставить вам как можно больше важных вспомогательных файлов. В большинстве случаев, последние версии библиотек и другие пакеты мы включаем в каталог «Essentials» (Главное) на прилагаемом диске. Поэтому, если в вашей системе возникли проблемы с зависимостями, следует заглянуть именно туда.

ФОРМАТЫ ПАКЕТОВ

Мы стараемся включать как можно больше различных типов установочных пакетов: RPM, Deb или любые другие. Просим вас принять во внимание, что мы ограничены свободным пространством и доступными бинарными выпусками программ. По возможности, мы будем включать исходные тексты для любого пакета, чтобы вы смогли собрать его самостоятельно.

ДОКУМЕНТАЦИЯ

На диске вы сможете найти всю необходимую информацию о том, как устанавливать и использовать некоторые программы. Пожалуйста, не забывайте, что большинство программ поставляются вместе со своей документацией, поэтому дополнительные материалы и файлы находятся в соответствующих директориях.

ЧТО ЭТО ЗА ФАЙЛЫ?

Если вы новичок в Linux, вас может смутить изобилие различных файлов и расширений. Так как мы стараемся собрать как можно больше вариантов пакетов для обеспечения совместимости, в одном каталоге часто находятся два или три файла для различных версий Linux, различных архитектур, исходные тексты и откомпилированные пакеты. Чтобы определить, какой именно файл вам нужен, необходимо обратить внимание на его имя или расширение:

имя_программы-1.0.1.i386.rpm – вероятно, это бинарный пакет RPM, предназначенный для работы на системах x86;

имя_программы-1.0.1.i386.deb – такой же пакет, но уже для Debian;

имя_программы-1.0.1.tar.gz – обычно это исходный код;

имя_программы-1.0.1.tgz – тот же файл, что и выше по списку: «tgz» - это сокращение от «tar.gz»;

имя_программы-1.0.1.tar.bz2 – тот же файл, но сжатый bzip2 вместо обычного gzip;

имя_программы-1.0.1.src.rpm – также исходный код, но поставляемый как RPM-пакет для упрощения процесса установки;

имя_программы-1.0.1.i386.FC4.RPM – бинарный пакет RPM для x86, предназначенный специально для операционной системы Fedora Core 4;

имя_программы-1.0.1.ppc.Suse9.rpm – бинарный пакет RPM, предназначенный специально для операционной системы SUSE 9.x PPC;

имя_программы-devel-1.0.1.i386.rpm – версия для разработчиков.

Если диск не читается...

Это маловероятно, но если все же прилагаемый к журналу диск поврежден, пожалуйста, свяжитесь с нашей службой поддержки по электронной почте: disks@linuxformat.ru

DVD

Сторона 1

Аудио
Amarok - самый функциональный в мире аудиоплеер
Banshee - аудиоплеер с каталогизатором музыки
BMPx - компактный аудиоплеер
Rhythmbox - аудиоплеер для среды GNOME

Рабочий стол
Gimmage - простой просмотрщик изображений
gjots2 - простой блокнот
GtkSort - сортировщик больших объемов данных
Kino - это DV-редактор для нелинейной обработки видео
Medit - текстовый редактор на GTK
Nero - коммерческая программа для записи CD/DVD-дисков
Scribus - открытая программа для профессиональной верстки
Tasks - приложение для управления списком дел
Wyneken - контент-ориентированный текстовый процессор

Разработка
Anjuta - гибкая среда разработки
Mono - реализация .NET для Unix
MonoDevelop - свободная среда разработки для Mono
TenDRA - компилятор для языков C и C++
Vim - продвинутый текстовый редактор

Дистрибутивы
Gentoo 2007.0 - дистрибутив, который может быть оптимизирован для любых задач
KDE_4_Live - живой дистрибутив с первой бета-версией KDE4
Symphony OS 2007 - живой дистрибутив с необычным рабочим столом.

Игры
DEFCON Demo - демо-версия новой игры от Introversion
Epiphany - это клон классической игры "диггер"
X-Moto - двумерный мотосимулятор

Справка
Ответы на часто задаваемые вопросы
LDP - часть архива документации Linux
RUTE - книга по системному администрированию Linux

Hot_Picks
CheckGmail - апплет для уведомления о новой почте
ClaSS - CMS для преподавателей
Eqonomize! - KDE-приложение для управления финансами
K3DSurf - программа для визуализации математических моделей
KTorrent - BitTorrent-клиент для среды KDE
Planets - простая интерактивная программа для имитации планетарной системы
PowerTOP - анализатор загруженности процессора
PyAwale - игра с камешками
SQLiteMan - графический интерфейс для SQLite3
Symbolica - головоломка

Интернет
Bitswash - кроссплатформенный клиент BitTorrent
Empathy - jabber-клиент и библиотека
GAdminHTTPD - интерфейс для управления веб-сервером Apache
KTrafficAnalyzer - анализатор трафика
Scythia - FTP-клиент на Qt4

Офис
AbiWord - текстовый процессор
Gnumeric - электронная таблица
KOffice - офисный пакет для среды KDE
OpenOffice.org - популярный офисный пакет

Система
DiskSearch - утилита для поиска файлов на сменных носителях
Disk manager - простой менеджер разделов
GParted - редактор разделов
Paco - это пакетный менеджер
SystemRescueCD - LiveCD для восстановительно-ремонтных работ
tgauge - утилита для отображения вывода lm_sensors

Сторона 2

Linux-Gamers LiveDVD 0.9.2

Комментарии? Присылайте ваши мысли и предложения по электронной почте: info@linuxformat.ru
Пожалуйста, ознакомьтесь с опубликованной в журнале инструкцией перед использованием данного диска.
Настоящий диск тщательно тестировался и проверялся на всех стадиях производства, однако, как и в случае с любым новым ПО, мы рекомендуем вам использовать антивирусный сканер. Мы также рекомендуем всегда иметь под рукой актуальную резервную копию данных вашего жесткого диска. К сожалению, редакция Linux Format не может принимать на себя ответственность за любые повреждения, разрушения или иные убытки, которые может повлечь за собой использование этого DVD, представленных на нем программ или данных. Перед тем, как устанавливать какое-либо ПО на компьютер, подключенный к сети, проконсультируйтесь с сетевым администратором.

Дефектные диски. В маловероятном случае обнаружения дефектов на данном диске, пожалуйста, обращайтесь по адресу: disks@linuxformat.ru

Тираж изготовлен на Уральском электронном заводе., 620066, Россия, г. Екатеринбург, ул. Комвузовская 17-203, Лицензия МПТР России ВАФ № 77-15

Поставляется вместе с журналом LINUXFORMAT номер 9(96) Сентябрь 2007

Создание установочных дисков при помощи CDRECORD

Самый быстрый способ записать ISO-образ на чистую матрицу – это *cdrecord*. Для всех перечисленных ниже действий потребуются права root. Для начала определите путь к вашему устройству для записи дисков. Наберите следующую команду:

```
cdrecord -scanbus
```

После этого на экране терминала должен отобразиться список устройств, подключенных к вашей системе. SCSI-адрес каждого устройства представляет собой три числа в левой колонке, например, 0,3,0. Теперь вы можете с легкостью записать образ на диск:

```
cdrecord dev=0,3,0 -v /путь к образу/image.iso
```

Чтобы упростить дальнейшее использование *cdrecord*, сохраните некоторые настройки в файле **/etc/default/cdrecord**. Добавьте по одной строке для каждого устройства записи (вероятно, в вашей системе присутствует всего одно такое устройство):

```
Plextor= 0,3,0 12 16M
```

Первое слово в этой строке – это метка, затем, после адреса SCSI-устройства вы должны указать скорость и размер буфера. Теперь вы можете заменить SCSI-адрес в командной строке на выбранную вами метку. Все будет еще проще, если вы добавите следующее:

```
CDR_DEVICE=Plextor
```

Все, что вам теперь нужно для записи ISO-образа – это набрать команду

```
cdrecord -v /path/to/image.iso
```

Если вы не из числа любителей командной строки, в таком случае вам придет на помощь утилита *gcombust*. Запустите ее из-под root, выберите вкладку Burn и ISO 9660 Image в верхней части окна. Введите путь к образу, который вы хотите записать на диск, и смело нажимайте на Combust!. Пока ваш образ пишется на диск, можете выпить чашечку кофе.

Другая ОС?

Вам не обязательно использовать Linux для записи компакт-диска. Все необходимые файлы уже включены в ISO-образ. Программы вроде *cdrecord* просто перенесут данные на чистую матрицу. Если у вас нет устройства для записи дисков, можно найти того, у кого оно есть, и записать диск на его компьютере. На нем может стоять Windows, Mac OS X, AmigaOS, или любая другая ОС.

Нет устройства для записи дисков?

А что, если у вас нет устройства, с помощью которого можно было записать образ на диск? Вы знаете кого-либо с таким устройством? Вам не обязательно использовать Linux для записи дисков, подойдет любая операционная система, способная распознать пишущий привод (см. выше).

Некоторые дистрибутивы умеют монтировать образы дисков и выполнять сетевую установку или даже установку с раздела жесткого диска. Конкретные методы, конечно, зависят от дистрибутива. За дополнительной информацией обращайтесь на web-сайт его разработчика. LXF

Содержание DVD

СТОРОНА 1

ЖУРНАЛ
JavaEE ..... Код примеров статьи.
Mono ..... Код примеров статьи.
PDFs ..... Подборка учебников GTK+.
RADIUS ..... Пример файла конфигурации

ЗВУК
Amarok ..... Музыкальный плейер для KDE.
Banshee ..... Музыкальный плейер для Mono.
BMPx ..... Beep Media Player Experimental.
Rhythmbox ..... Музыкальный плейер для Gnome.

РАБОЧИЙ СТОЛ
Gimmage ..... Программа просмотра изображений на основе Gtk.
gjots ..... Обработка заметок.
GtkSort ..... Манипулятор данных.
Kino ..... Видеоредактор.
Medit ..... Текстовый редактор.
Nero Demo ..... Программа для записи дисков.
Scribus ..... Настольная издательская система.
Tasks ..... Менеджер списка задач.
Wyneken ..... Контент-ориентированный текстовый процессор.

РАЗРАБОТКА
Anjuta ..... Интегрированная среда разработки.
Mono ..... Среда .NET с открытым кодом.
MonoDevelop ..... Интегрированная среда разработки C#.
Tendra ..... Компилятор C/C++ с лицензией BSD.
Vim ..... Vi iMproved (усовершенствованный Vi).

ДИСТРИБУТИВЫ
Gentoo ..... Дистрибутив для бывалых.
KDE 4 Live ..... Предварительный релиз KDE 4.
Symphony OS ..... Live-дистрибутив на базе Mezzo.

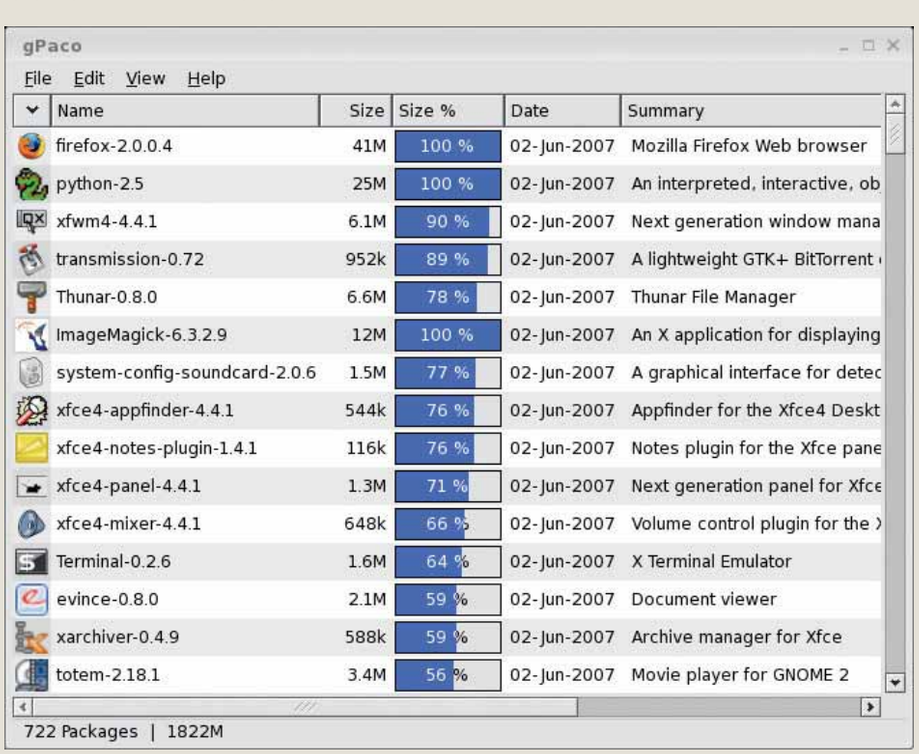
ИГРЫ
Defcon Demo ..... Имитатор ядерной войны.
Epiphany ..... Клон Boulderdash.
X-Moto ..... Мотокросс.

СПРАВКА
Answers ..... Решенные проблемы Linux.
LDP ..... Linux Documentation Project.
RUTE ..... Руководство по администрированию Linux.

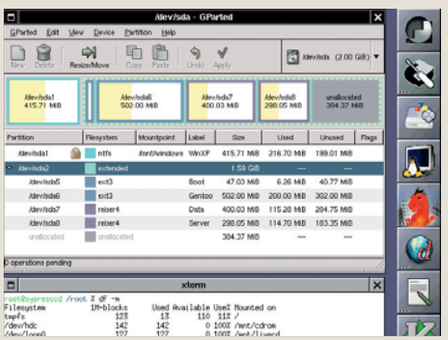
HOTPICKS
CheckGmail ..... Программа уведомления о новых сообщениях Gmail.
ClaSS ..... Программа для учителей.
Eqonomize ..... Персональный бухучет.
K3DSurf ..... Визуализация трехмерных поверхностей.
KTorrent ..... Клиент BitTorrent для KDE.
Planets ..... Орбитальный симулятор.
PowerTOP ..... Монитор энергопотребления.
PyAwale ..... Игра Awale по правилам Берега Слоновой кости.
SQLiteMan ..... Графическая оболочка для SQLite.
Symbolica ..... Порт головоломки C64.

ИНТЕРНЕТ
Bitswash ..... Клиент BitTorrent.
Empathy ..... Программа обмена сообщениями.
GAdminHTTPD ..... Инструмент настройки Apache.
KTrafficAnalyzer ..... Монитор трафика.
Scythia ..... FTP-клиент на базе Qt.

ОФИС
AbiWord ..... Текстовый процессор.
Gnumeric ..... Редактор электронных таблиц Gnome.
KOffice ..... Офисный пакет KDE.
OpenOffice.org ..... Крупный офисный пакет.
Siag ..... Ультралегкий офисный пакет.



› ***Paco*** **протягивает руку помощи в создании, установке и управлении программами.**



› **System Rescue CD: запишите диск и загружайтесь с него, чтобы восстановить систему Linux после серьезного сбоя.**

СИСТЕМА
DiskSearch ..... Поиск по съемным носителям.
Disk Manager ..... Конфигуратор файловой системы.
GParted ..... Редактор разделов Gnome.
Paco ..... Система управления пакетами.
System Rescue CD ..... Живой дистрибутив для восстановления Linux.
tgauge ..... Монитор температуры CPU.

ГЛАВНОЕ
ATI driver ..... Графический драйвер.
Bash ..... Оболочка командной строки.
Cairo ..... Библиотека двумерной графики.
CheckInstall ..... Программа создания двоичных пакетов.
Coreutils ..... Утилиты командной строки.
GLib ..... Низкоуровневая библиотека.
Glibc ..... Библиотека GNU C.
GTK ..... Инструментарий пользовательского интерфейса.
HardInfo ..... Системная информация и сравнительный анализ.
Jigdo ..... Создатель ISO-образов.
Kernel ..... Свежий релиз ядра Linux.
LibXML2 ..... XML-анализатор и набор инструментов.
Ncurses ..... Оконный инструментарий текстового режима.
NVIDIA driver ..... Графический драйвер.
Python ..... Язык программирования.
Rawrite ..... Программа записи образов на дискеты.
SBM ..... The Smart Boot Manager.
SDL ..... Библиотека мультимедиа.

СТОРОНА 2

LinuX-gamers LiveDVD ..... Сборник свободных игр для Linux

Лучшие новинки открытого ПО на планете

LXF HotPicks

Ник Вейч
Чудо-математик, фанатик Flickr, иногда редактор, Ник из высокой башни LXF краем глаза осматривает поле свободных программ в поисках хорошеньких приложений.

В ЭТОТ РАЗ ТОЛЬКО ДЛЯ ВАС: *KTorrent* » Planets » K3Dsurf » Eqonomise » Powertop » CheckGmail » ClaSS » Sqliteman » Pyawale » Symbolica

Клиент Bittorrent

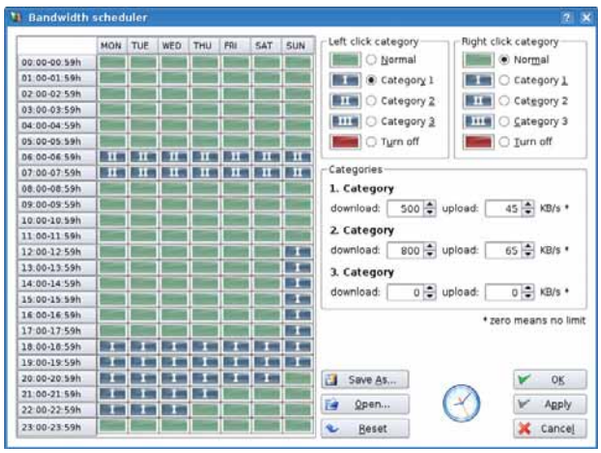
KTorrent

Версия 2.2 **Web** www.ktorrent.org

Мы не станем вникать, зачем и для чего люди используют торрент-клиенты. Все знают, что их использование может быть законным и незаконным – то же касается и большинства других приложений (а если вы не верите, что это относится и к *Crack Attack!*, поглядите, как Майк и Пол играют в него). Не будь торрент-клиентов, вы не получали бы каждый месяц самые свежие версии дистрибутивов на дискеприложении. Оставим за бортом моральные аргументы по поводу загрузки телепрограмм, которые вы могли записать прошлой ночью (полагаем, ваш магнитофон не содержит проприетарных ограничений в прошивке, мешающих запрограммировать его соответствующим образом...).

На свете есть великое множество торрент-клиентов для Linux (мы публиковали их сравнение в LXF90), но не думайте, что все они на одно лицо. Они отличаются несколькими ключевыми особенностями, а у *KTorrent* особенностей немало. Например, в нашем обзоре победил *Azureus*, но он работает под Java и требует намного больше ресурсов компьютера, чем *KTorrent*.

Кроме того, с момента последнего Сравнения произошли значительные изменения, особенно в интерфейсе пользователя. В старом варианте для каждого вида деятельности имелась своя вкладка, но теперь все торренты могут отображаться в одной таблице – включая те, что уже загружены и теперь лишь раздаются. А по умолчанию у вас две вкладки: одна для загрузок, а другая для раздачи – так более понятно. В любом случае, организация торрентов стала намного проще с добавлением функции группировки, так что просмотр загрузок вашего дистрибутива и других файлов стал намного легче.

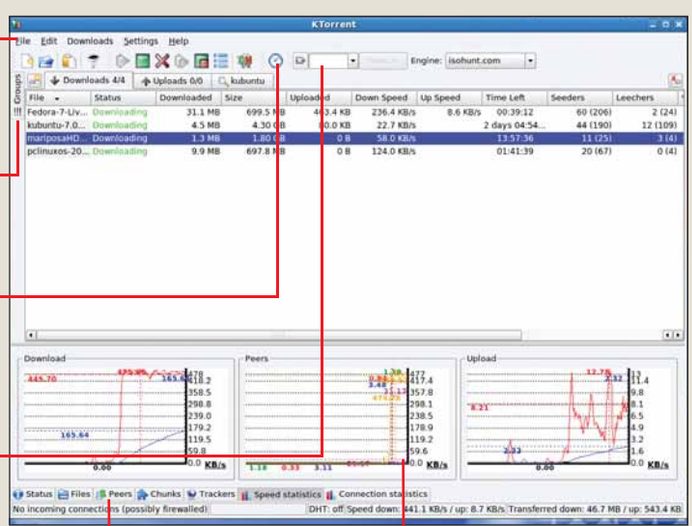


› Модуль управления шириной канала облегчает приоритизирование ваших публичных файлов, одновременно используя интернет-соединение для других задач.

KTorrent поддерживает все последние технологии мира торрент-клиентов, включая DHT для поиска дополнительных узлов. Система расширений теперь включает множество полезных элементов, включая RSS-читалку (для мгновенного скачивания новых торрентов по мере их появления – разве не полезно знать, когда в эфир выходят Ubuntu-торренты, чтобы автоматически подцепить их?). Кроме того, теперь имеется модуль web-интерфейса – запустите *KTorrent* на своем домашнем сервере, и сможете получить доступ к нему отовсюду. Это удобно для усечения ширины канала, когда вы хотите заняться в Интернете чем-то помимо раздачи файлов. Вы можете возразить, что некоторые из этих функций заходят слишком далеко (например, расстановка приоритетов в наборе торрентов), но все они, в целом, ненавязчивы. Интерфейс пользователя все так же экономен и понятен, при всей таящейся в нем мощи, и *KTorrent* по-прежнему проворное и отзывчивое приложение.

Несомненно, будет еще много версий целого спектра торрент-клиентов – похоже, на данный момент это область некоторого обострения интереса – но если вы не попробовали *KTorrent* после нашего старого Сравнения, попробуйте сейчас.

Исследуем интерфейс *KTorrent*



Действуйте сами
Создайте свой собственный торрент в меню файлов. Это очень просто.

Группировка файлов
Управляйте множеством файлов, объединив их по категориям.

Следуйте расписанию
Активируйте модуль диспетчирования для управления шириной канала.

Поиск торрентов
Поиск свяжет с популярными торрент-сайтами, а если хотите, может открыть их в стандартном браузере.

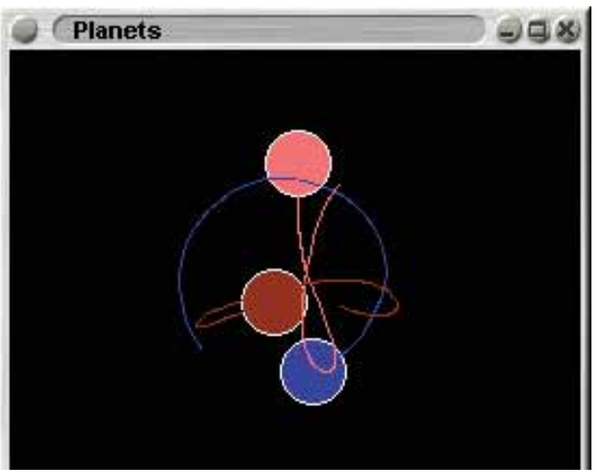
Проверка статуса
Проверьте, к каким узлам всемирной сети вы подключились.

Проверка соединения
Следите за использованием канала при помощи модуля статистики.

Физический имитатор

Planets

Версия 0.1.10 **Web** http://planets.homedns.org



› Создайте свои собственные странные аттракторы.

Космос велик – довольно трудно запихать его в компьютер с целью по нему разгуливать. Что, несомненно, и послужило стимулом для создания этой небольшой игрушки – графического имитатора физических законов, который приобщит вас к тайнам планетарной механики, хоть вы и не покоритель пространства.

Планеты добавляются в среду простым нажатием клавиши A. Новое небесное тело получает цвет шербета и немедля плюхается на экран. Остается придать ему начальную скорость и запустить (щелкните на нем и нарисуйте вектор) – теперь только вы его и видели. При вашей первой попытке планета честно отправится в заданном направлении и тут же исчезнет за границей экрана. Вы можете сделать две вещи: или увеличить масштаб при помощи клавиши – и продолжать уменьшение до тех пор, пока планета не превратиться в крошечный пиксель, или нажать клавишу P для паузы и попытаться внести небольшое оживление, добавив другие планеты.

При известном везении, добавка пары планет создаст более стабильную систему, и вы сможете несколько минут в восторге наблюдать, как пульсирует в свечении лавы ваша новая вселенная, пока все её комочки не сольются или не разлетятся в бесконечность. Если вы хотите получить что-то действительно причудливое, включите режим «отскок», и планеты превратятся в гравитационные резиновые шары, непрерывно сталкивающиеся друг с другом при вращении в бесконечном пространстве (или ограниченном размерами экрана).

С другой стороны, отсутствует возможность реального ввода данных – вы можете лишь добавлять планеты, и все. Неплохо было бы, например, уметь изменять размер/массу объектов (единственный способ сделать это – изменить масштаб перед их созданием), чтобы создавать более сложные системы. Возможность числового ввода начальных векторов также была бы немалым подспорьем для творцов новых миров.

> «В чем же смысл всего этого? А никакого смысла и нет!»

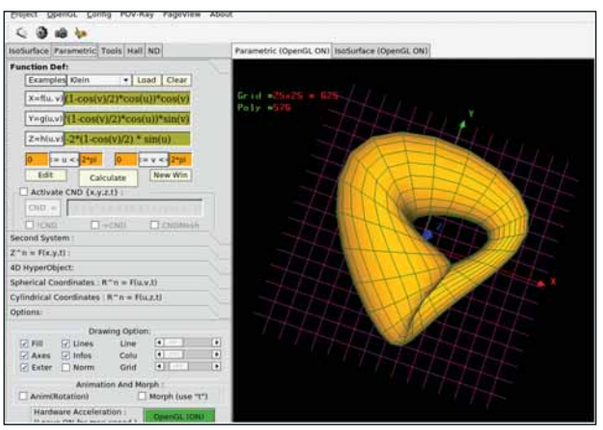
В чем же смысл всего этого? А никакого смысла и нет! Хотя есть нечто очищающее в создании малых миров и наблюдении за тем, как они беззащитно вращаются по вашей прихоти, пока не придут к некой стабильности, а затем сольются в одну большую планету, и можно раскручивать их вновь.

Как бы там ни было, если вам удастся создать точную работающую копию известной Вселенной, то напишите и расскажите нам.

Визуализатор 3D-поверхностей

K3Dsurf

Версия 0.6.1 **Web** http://k3dsurf.sourceforge.net



› Если вы уронили бутылку Клейна в лесу, когда некому было это услышать...

Чарли не может кататься на серфинге, но вы-то можете – так берите шлепанцы, полотенце... Э, погодите-ка. Похоже, что этому небольшому приложению нечего делать на волнорезах и пляжах, но то, с чем оно имеет дело, куда более удивительно. Математика!

Вы все еще здесь? На самом деле, математика – это забавно. По крайней мере, когда вместо того, чтобы пялиться на ряды уравнений, можно просматривать их, упаковав в несколько измерений с анимацией и тенями. *K3DSurf* использует мощь OpenGL для представления фантастических видов (а если ваша видеокарта не поддерживает OpenGL, вы все равно добьетесь цели, используя программную отрисовку).

Установите это легковесное приложение, запустите – и вы обнаружите небольшое окно с квадратной областью отображения и панелью управления с закладками слева. Щелчок на вкладках откроет различные способы ввода уравнений, а также основные опции цвета и вида и всякие оформительские штучки.

Я понимаю, что у отдельно взятых читателей LXF может и не быть в голове полчищ 3D-уравнений, рвущихся на ввод, но не пугайтесь: чтобы дать вам войти во вкус, предусмотрены готовые примеры (с ними разрешается играть). Встроенные примеры, помогающие приступить к делу, есть во всех режимах построения; доступ к ним можно получить при помощи выпадающего меню каждой управляющей вкладки. Если вам охота поэкспериментировать с чем-нибудь самостоятельно (оно и рекомендуется), можете просто щелкнуть в текстовом поле для изменения уравнения или создании собственного. Пределы функции показаны внизу, и работа с ними также весьма поучительна! По завершении редактирования каждый раз нажимайте на кнопку Generate [Сгенерировать], чтобы обновить экран.

> «Если для вас это трудно, можете также отрисовать в POVRay.»

Можно сохранять экранный снимок текущего объекта в окне, а можно экспортировать сгенерированные формы в вездесущий формат Wavefront OBJ для дальнейшего использования. Но если это кажется трудоемким, можете также выполнить отрисовку при помощи движка POVRay 3D, прямо из программы, повысив качество результата.

Реальное применение данного приложения – визуализация математических формул, но это же просто прекрасная игрушка для забавы! Изучите все вкладки и наиграйтесь со встроенными уравнениями, создавая прекрасные вариации. Бесконечность – внутри!

Домашние финансы

Eqonomise

Версия 0.5 **Web** http://eqonomize.sourceforge.net

Аеньги, вернитесь! Клянусь, я сберегу вас... ой, простите. Итак, перед вами персональный финансовый менеджер – он подводит баланс вашей чековой книжки и предоставляет подробную статистику, куда ушло все нажитое непосильным трудом (по ходу, советуем рассмотреть наше предложение о подписке: это намного дешевле, чем покупать в розницу каждый месяц). На этой арене уже есть несколько соперников, но если честно, то с большинством из них вы ох как намаетесь. Если вы хотите получить примерное представление о своих финансах, не продираясь через 20 различных экранов, испещренных мелкими цифрами, то это приложение для вас.

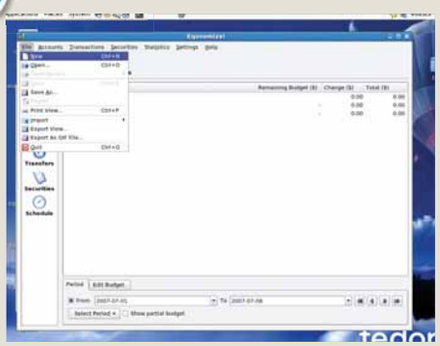
Интерфейс опрятный и минималистичный, как кухня спартанца после весеннего авраала. Вкладок с опциями не так уж много, так что не бойтесь запутаться: несколько иконок сбоку, для переключения формы представления, и простое меню для главнейшего – ввода информации.

> «Интерфейс опрятный и минималистичный, как кухня спартанца.»

Любители поиграть на бирже могут также добавить информацию об акциях в специальный раздел приложения. Это даст вам удобное представление о ваших фондах (или их отсутствии) в каждый конкретный момент времени.

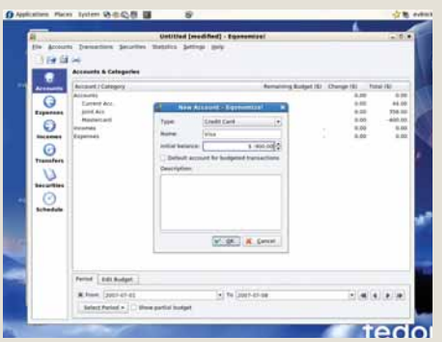
Приложение не так чтобы набито функциями до отказа, но это, может, и к лучшему: легко понять, как оно работает (весьма кстати, поскольку документации нет), и вы можете оценить свои финансы почти что в 0 секунд.

Шаг за шагом: Составьте бухгалтерский баланс



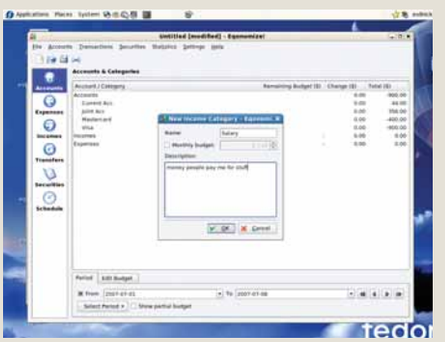
Начнем сначала!

Выберите New [Создать] в меню File [Файл], чтобы начать с нуля. Затем выберите пункт меню New Account [Новый счет] и введите данные текущего банковского счета, которым хотите управлять.



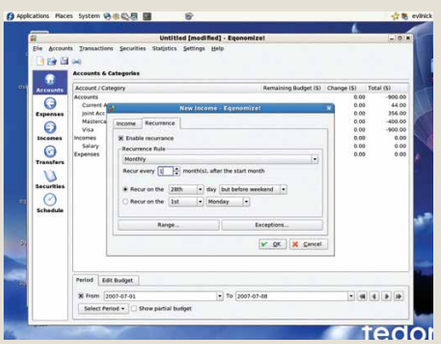
Добавим счета

Введите еще счета для наличных, кредитных карт и так далее. Добавляйте текущий баланс для каждого счета при его создании – потом это сильно облегчит управление!



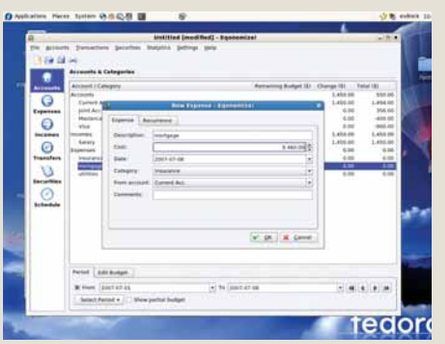
Создадим доходы

Теперь вам надо обзавестись доходами! Это может быть выигрыш в лотерею или в покер, зарплата, гонорар за вашу книгу-бестселлер – да что угодно. Выберите Accounts > New Income Category [Счета > Новая категория доходов] для их создания.



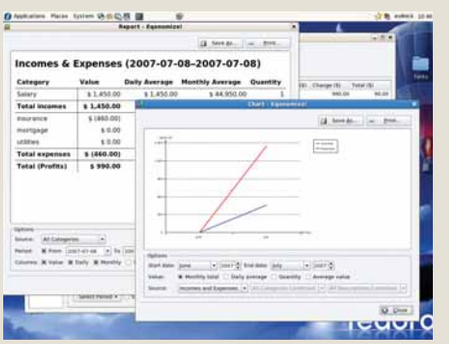
Добавим транзакции

Настроим стандартные транзакции для доходов. Выберите Transactions > New Income [Транзакции > Приход] и введите детали. Щелкните на вкладке Recurrence [Повторение], чтобы настроить повторяющиеся события для регулярных поступлений.



Добавим расходы

Как ни грустно, но расходы также случаются. Вновь выберите Accounts > New Expense category [Счета > Новая категория расходов] и повторите действия. Добавьте повторяющиеся события для квартплаты или выплат по кредитам – чем больше автоматизации, тем лучше.



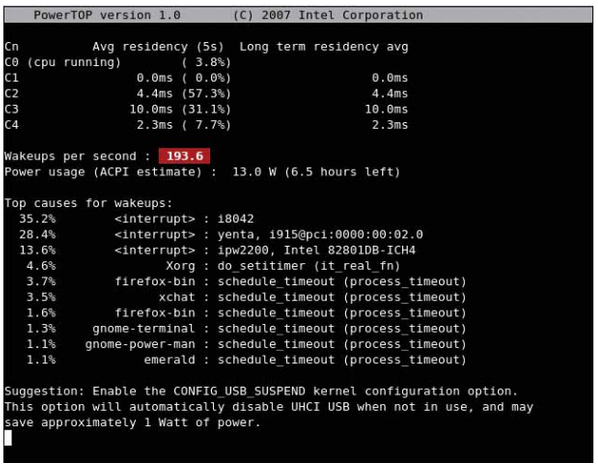
Получаем обновления в графике

Ну вот, полюбуйтесь, как плохо у вас с финансами! Выберите опцию из меню статистики, чтобы увидеть всякие диаграммы, отображающие направление вашего финансового краха!

Советник по энергосбережению

Powertop

Версия 1.7 **Web** www.linuxpowertop.org/index.php



› Найти и убить прожорливые до ресурсов процессы.

Ноутбуки – это здорово! Берете его подмышку, шляетесь по всему миру, открываете его где-то вблизи точки Wi-Fi доступа и проверяете, попивая кофе, какую там ерунду Майк запихал в блог **LXF**! Тут, не дав вам дочитать забавную опубликованную перепалку, батарея дохнет, и вы остаетесь с тяжелым кирпичом, который должны таскать с собой, пока не найдете электрическую розетку. Нет, ноутбуки – дрянь.

Одна из проблем (не говоря о плохой поддержке проприетарных технологий энергосбережения) – избыточные обращения: состояние простоя не продолжается достаточно долго, потому что процессор пробуждается тем или иным приложением, даже когда оно ничего не делает. А данное приложение (разработанное Intel и выпущенное по лицензии GPL) – это спаситель жизни батарей!

Powertop напоминает *top* (приложение, наблюдающее за процессами). Но вместо слежения за использованием процессора и памяти, *Powertop* анализирует количество активаций CPU приложениями. Очевидно, многие из пробуждений связаны с некорректным (или по крайней мере неэффективным) использованием таймеров. Знание приложений, вызывающих проблемы, означает, что вы можете подправить их (если вы разработчик или одаренный любитель), заменить аналогичными приложениями без проблем или попробовать обходные пути, описанные на сайте *Powertop* – там таких много. Много там и советов по расшифровке результатов, поскольку иногда приложение, выглядящее пожирателем ресурсов, на самом деле таковым не является, а беда во взаимозависимой природе большинства сложных приложений.

Небольшое предупреждение: чтобы данное приложение работало, у вас должно быть ядро версии не ниже 2.6.21 с включенной функцией dynticks! Возможно, придется компилировать ядро, поскольку к моменту написания большинство дистрибутивов не имели даже доступных обновлений до этой версии. Однако сборка достаточно проста. На сайте также есть список того, какие опции следует включить или отключить при компиляции ядра, так что вам следует его посетить.

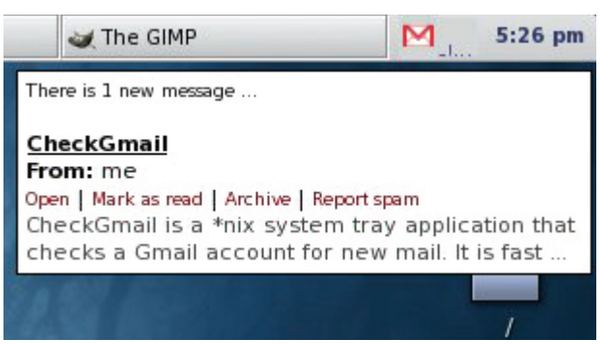
> «Powertop анализирует количество активаций CPU приложениями.»

Но помните, что вы не обязаны постоянно использовать эту версию ядра, поскольку *Powertop* – это лишь помощь в обнаружении приложений с плохим поведением в вашей системе: как только вы засечете проблемы, можете вернуться обратно к вашему нормальному ядру и наслаждаться долгожительством батарей. Каждый миллиампер-час на счету!

Уведомление о приходе почты

CheckGmail

Версия 1.12 **Web** http://checkgmail.sourceforge.net



› Вы никогда больше не пропустите письма.

Остался ли кто на планете, не имеющий учетной записи на Gmail? Подозреваем, что только люди, работающие с Yahoo и Hotmail (и даже они, вероятно, потихоньку его завели). Но хотя Gmail очень мил – ну как узнать, что к вам пришла ценная почта из некоего банка с запросом вашего имени пользователя и пароля по «соображениям безопасности», если не держать его открытым постоянно? Такие вопросы писатели любят называть риторическими, а теперь мы продолжим, полагая, что вы мысленно дали верный ответ (или, возможно, выкрикнули его, потрясая журналом над головой).

Если только вы по жизни никогда не работали на компьютере, и не держите окно браузера постоянно открытым, вы не можете утверждать, что видите вспышку вкладки Gmail при получении заветного письма от вашей воображаемой девушки.

CheckGmail – небольшая утилита, написанная на Perl для Unix-систем. Она смирно сидит в системном лотке и компенсирует тот факт, что несмотря на все свое величие, Google предусмотрел такую вещь только для других платформ. Хммм. Как бы то ни было, перед переходом к попыткам установить и использовать это приложение вам следует знать, что оно требует целую кучу модулей Perl – но ведь Perl-приложениям всем такое подавай. Ему также нужна потоковая версия Perl, нетипичная для дистрибутивов вплоть до последних шести месяцев или около того, поскольку старая поточная модель обладала неплохой производительностью. К счастью, запуск скрипта из оболочки сообщит вам, есть ли какие-то проблемы с наличным Perl и каких модулей не хватает (внимание: хотя некоторые из них помечены в readme как необязательные – вроде различных модулей шифрования – мы обнаружили, что они важны, и определенно рекомендуются). Когда вы заставите его работать, на лотке появится иконка.

> «Оно требует кучи модулей Perl, но Perl-приложениям всем такое подавай.»

В любом случае – установив его, вы сразу же сможете заметить, когда пришло новое письмо. Того лучше: умело придержав мышь над соответствующей областью, вы сможете прочитать сводку о последних 20 сообщениях, и даже перейти по внедренным ссылкам. Однако кое над чем необходимо поработать, по крайней мере с версией, установленной в Fedora, поскольку шрифты ведут себя немного странно – правда, быстро исправляются в настройках...

Конечно же, для Linux имеется много других инструментов опроса Gmail, но этот разумно мал и прост, а предпросмотр писем при работе хорош. Весьма рекомендуем попробовать.

Управление учебным классом

ClaSS

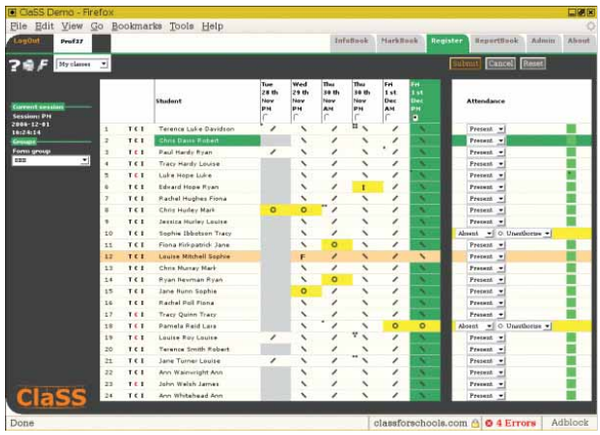
Версия 0.8.20 **Web** www.laex.org/class

Мы не часто рассматриваем специализированные проекты на наших священных страницах. Ну, кроме математических, астрономических и инструментов для цифровых камер. Тем не менее, данное образовательное приложение весьма заслуживает упоминания. Пишем – и прямо слышим, как вы скучливо закрываете страницу; но задержитесь на минутку. Это не некая система обучения началам математики или клавиатурный тренажер – это настоящее приложение для школ. *ClaSS* – основанная на web система администрирования для учителей. Ее цель – чуть-чуть упростить введение в рамки борьбы с маленькими нахалятами (кто сомневается, что образование – это дело честолюбцев и общественных лидеров завтрашнего дня?), путем автоматизации множества административных задач при помощи одного простого web-приложения.

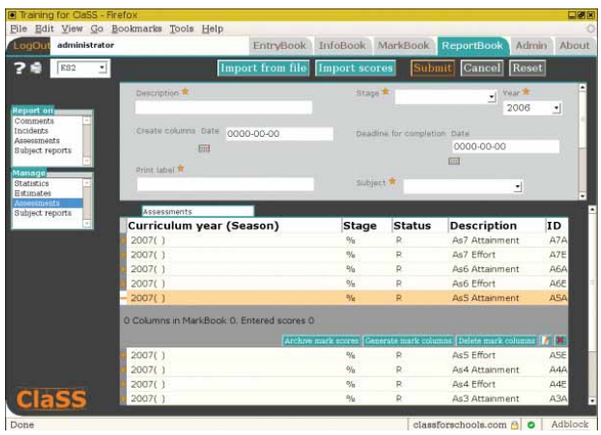
ClaSS следит за учениками, посещаемостью, оценками, занятиями и любыми замечаниями через несколько вкладок вверху главной страницы. Войдите как учитель, и сможете быстро перемещаться по web-сайту, чтобы добавить информацию об ученике, обновить ежедневный журнал, проверить текущие успехи или ускользнуть в комнату отдыха, чтобы выпить заслуженную чашку кофе со сливками. Вообще-то мы не смогли найти последнюю опцию в разделах сайта, но она точно где-то есть.

Поскольку это web-приложение, вам едва ли потребуется больше, чем отдельный сервер, за которым можно сидеть, и при разумном финансировании, его можно скооперировать с учителями других школ – хорошая тема для разговора о классе 3G на выездном уроке по географии.

ClaSS никоим образом не является полной школьной системой (здесь, например, нет функций для самих учеников), но по крайней мере предоставляет основные службы для преподавательского состава. Если вы не хотите мучиться с его установкой, чтобы просто посмотреть, подходит ли оно для вашей школы, то на сайте имеется демо-версия. Только не обращайте внимания на скорость работы – несомненно, на вашем локальном сервере *ClaSS* будет работать куда быстрее.



› **Перекличка еще никогда не была такой веселой.**



Менеджер баз данных

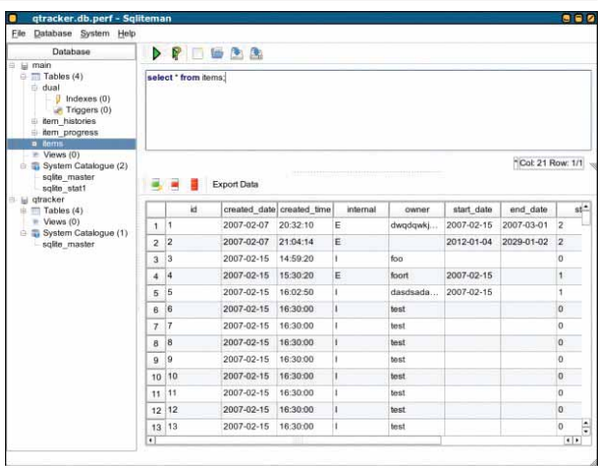
Sqliteman

Версия 0.99 **Web** http://sqliteman.sourceforge.net

Sqlite весьма популярен. Базы данных – удобное место для хранения: если, скажем, у вас много книг, то вам хочется поместить их на книжную полку; если же у вас много данных, то вы помещаете их в базу данных. Это вполне очевидно. Но что если вам необходимо пристроить лишь пачку брошюр, а не разваливающиеся кучи толстых томов? Загромождение пространства огромным книжным шкафом кажется чрезмерным. Так же и с данными. *Sqlite* был задуман для того, чтобы предоставить приложениям возможность хранить данные знакомым и удобным способом SQL, но без мороки с запуском сервера баз данных и всяким там администрированием, настройкой и прочими неприятностями.

Сейчас управление книжными полками, ну, я имею в виду, базами данных, является – для большинства людей достаточно удачливых, чтобы не страдать крайней формой ОКР – утомительной и требующей времени работой. А как обстоит дело с управлением файлами *Sqlite*? Можем ли мы подправить данные, если нужно, или просто управлять файлами понятным и простым способом? Один из возможных ответов – *Sqliteman*, оболочка с графическим интерфейсом, созданная для управления данными в файлах *Sqlite* простым и визуальным способом.

Эта версия использует интерфейс *Qt 4*, так что убедитесь, что он у вас установлен. Вероятно, вам понадобится переменная среды QTDIR, если вы компилируете программу из исходных текстов, потому что даже если Qt 4 установлен, многие дистрибутивы по умолчанию все еще используют *Qt 3*. Вообще-то сборка этого приложения может быть трудной, поскольку некоторые из зависимостей, похоже, не проверяются при конфигурировании, но в жизни такое бывает – это всего лишь релиз-кандидат, и если вы обнаружите ошибки, то авторы, вероятно, будут благодарны вам за отчет о них, но сначала убедитесь, что у вас последний SVN-срез.



› **Это база данных. Знаете, мы здесь не дурака валяем.**

> **«Sqliteman хорош и для разработчиков, и для пользователей.»**

Запустив его, вы можете вполне успешно манипулировать вашими данными различными способами: создать новую базу данных или открыть существующую, а затем поиграть с ней. Вид по умолчанию в левой панели содержит древовидное предоставление текущей базы данных; правая панель изменяется в зависимости от того, что вы выбираете. *Sqliteman* превосходен для разработчиков и тех, кто работает с данным форматом баз данных. Он не приспособлен для реального ввода данных, но все остальное (чего немало) работает отлично.

HotGames Развлекательные приложения

Стратегия на фишках

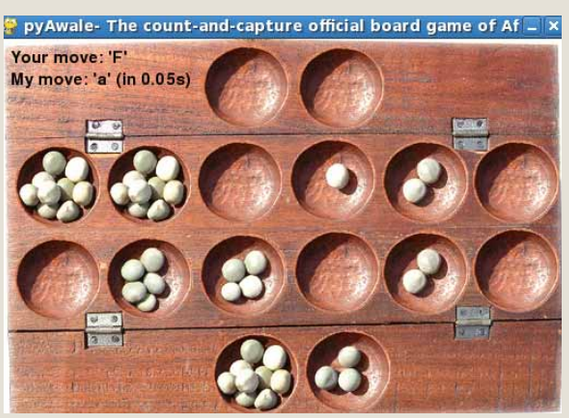
Pyawale

Версия 0.2.1 **Web** https://gna.org/projects/pyawale

Задолго до эры Nintendo DS или Donkey Kong, или даже крикета, люди сами себя развлекали. Вероятно, это были тараканьи бега или швыряние камней в Майка. Так или иначе, где-то в далеком прошлом люди однажды открыли огонь, и теперь вечером что-то надо делать – действительно надо что-то делать. Однажды, видимо, кто-то понабрал камней, нашел доску с дырками и изобрел незатейливую, но чертовски трудную игру. Ее, наверное, назвали '*Age of Empires*' [Эпоха империй], но тут же попали под суд за незаконное использование товарных знаков (в будущем, юристы MS изобретут путешествия во времени и межгалактические перелеты, угрожая людям, как далеки [Dalek – один из роботов в детской телепередаче, – прим. пер.], только без такого же типажного прикида). Имеется много разновидностей этой игры, и великое множество различных правил. Этот пример использует правила Берега Слоновой кости для разновидности Awale (официальной настольной игры в Африке). А именно, игроки делают ходы, чтобы забрать камушки из чашек (или «домов») перед ними. Затем внутренние камни по одному «высеиваются» в соседние чашки против часовой стрелки. Если последний камушек попадает в дом оппонента и доводит число камней в нем до двух или трех, камни считаются захваченными. Имеется множество сложных правил, узнать о них можно в документации к самой игре.

Для игры в графическом режиме необходимо также установить *pygame*.

› Если вы уже фанат Mancala или Bantumi, готовьтесь к огорчениям и развлечениям в равной степени...



Головоломка

Symbolica

Версия 0.8 **Web** http://sourceforge.net/projects/symbolica

Почему-то, сколько ни создавай новых прекрасных игр, всегда есть вневременная классика, которая видится нашим внутренним взором как сияющие вершины Игрландии. Кажется, что некоторые игры будут жить вечно. Но данная игра не из таких.

Symbolica – римейк старой игры, впервые появившейся на C64. Хотя она и не перевернет мир, но это все-таки достаточно интересная небольшая игра. Случайным образом из символов создаются цепочки. Каждый раз, когда вы делаете ход (кстати, вы – это мраморный шарик), символы меняются на соседний из списка, пока не будет достигнут конец и они не исчезнут. Идея в том, чтобы пройти по всем квадратикам, заполнив их все. Кажется, просто, да? Что ж, попробуйте! Музыка тоже хороша. Чтоб не забыть – имеется версия для GP2X (http://gp2x.co.uk), а также для планшетов Nokia... LXF

› Продуктивность вашей работы явно плоха!

Также выпущены

Новые и обновлённые программы, заслуживающие вашего внимания...

evilwm 1.0.0. Оконный менеджер, сведенный к минимуму, но все еще пригодный для использования!
http://www.6809.org.uk/evilwm/

Qtstalker 0.34. Приложение для слежения и анализа за фондовым рынком
http://qtstalker.sourceforge.net/

Kernbench 0.42. Инструмент тестирования скорости процессора
http://freshmeat.net/projects/kernbench/

Qemulator 0.5. Легкое управление задачами QEMU
http://qemulator.createweb.de/

GAG 4.9. Графический менеджер загрузки с продвинутыми функциями
http://gag.sourceforge.net/

HylaFAX 5.1.5. Последняя версия почтенного приложения для отсылки и приема факсов
http://hylafax.sourceforge.net/

NomachineNX. Клиент/серверный удаленный рабочий стол
http://www.nomachine.com

› Смотри на мир через nomachine!

Slackroll. Менеджер пакетов для Slackware http://slackroll.sourceforge.net

Decibel 0.04. Аудио-проигрыватель на базе GTK+, с простым интерфейсом
http://decibel.silent-blade.org

› Decibel – минимум аудио.

Warzone 2100 Resurrection 2.0.7. Отличная RTS-игра. Здесь танки, взрывы и прочее http://wz2100.net

Пингвины в уездном городе

Александр Михайлович Поносов

Оханск. Некогда – город, заправлявший делами огромного уезда, в том числе и Сепычевской волости, ныне тихий уютный районный центр на берегу Камы. Мысль посетить его появилась еще зимой, когда прочитал на форуме сообщение:

12 января 2007 13:01 Оханский ЦДТ Open Source

С сентября 2006 года мы в Оханском Центре детского творчества перешли в обучении информатике на Linux. (Оханск и Верещагино – соседи). Используем Mandriva Linux. Затраты на переход – несколько сотен рублей (стоимость носителей и их пересылки). Чудесная вещь! Все есть, что надо для работы, и даже программы, написанные для Windows и MS DOS, работают. Завели блог на эту тему: http://katet.ru/.

«Мы» – это братья Сибиряковы, Сергей и Александр. Познакомиться с ними, а также с директором ЦДТ Трушниковым Михаилом Прокопьевичем довелось только летом в первые дни отпуска. Впечатления самые добрые: замечательные люди, чудесная прикамская природа, море полезной информации, погода и дороги не в счет. Сергей Викторович оперативно поведал о «встрече союзников на Каме» в сетевом журнале.

На абордаж Windows

Вчера в Оханский Центр детского творчества заезжал на огонек светодиодов компов с Линуксом известный «пират» Александр Поносов с супругой.

Оханск встретил чету Поносовых июльской грозой с ливнем, пришедшими из-за реки Камы. Кстати, Кама известна тем, что исторически ошибочно считается притоком Волги, хотя в реальности все наоборот. :)

Встретились по-соседски, разговаривали часа два. Александр пишет статью на тему «Линукс в образовании» и его заинтересовал наш скромный опыт. Говорили о многом, сумбурно и эмоционально. Встреча записывалась Александром на диктофон.

Свежий воздух после грозы и нескольких дней тридцатиградусной жары способствовал активной беседе.

Похвастались нашей небольшой коллекцией раритетных бытовых компьютеров, показали игрушки, сделанные детьми из контрафактных CD.

В конце беседы стороны сошлись на том, что Линукс стоит продвигать в школы, и за этим стоит огромный потенциал.

Такой вот неожиданный итог встречи :)

Я же только недавно разобрал запись нашей «сумбурной, эмоциональной, активной» беседы, и предложу читателям ее небольшую часть с «вкраплениями» из публикаций Сергея Сибирякова в сети.

Александр Поносов (АП): **Вы давно в системе дополнительного образования?**

Сергей Сибиряков (СС): Еще с советских времен. Был директором станции юных техников, вел компьютерный кружок. В результате оптимизации СЮТ вошла в состав Центра детского творчества. Сейчас на пару с братом занимаемся клубом. Собственно, преподает Александр, а я решаю технические вопросы.

АП: **Как и когда появилась идея перейти на Linux?**

СС: Идея зрела давно. Во-первых, мы – учреждение дополнительного образования, поэтому можем более свободно выбирать темы и направления своей образовательной деятельности, чем средние школы. Мы творить должны по статусу, вот и вытворяем, что хотим. Ну что можно вытворять с Windows? Только в игрушки играть! А в Linux с этим похуже, зато его самого можно сделать игрушкой, загадкой, конструктором. В этой системе заложен огромный образовательный потенциал.

Нас интересует Интернет, и нам хотелось бы заниматься с детьми именно интернет-технологиями. А Интернет – это Unix. Пользоваться в Интернете Windows – все равно, что рассекать по морю на «Запорожце»: как ни конопать, все равно затонет.

Следующее: финансовые соображения. Денег на приобретение программного обеспечения у нас нет, так как Оханский район Пермского края – сельский район с большими проблемами, и бюджет у него – с гулькин нос. Развиваем направление научно-технического творчества только за счет благотворительности частных лиц и организаций, которой едва хватает на то, чтобы ремонтировать и обновлять понемножку «железо». Какие тут могут быть закупки ПО Microsoft? Выход: пиратство или переход на Open Source.

АП: **Значит, задумка давняя, а что было толчком к миграции?**

СС: В конце мая 2006 года мне позвонил прокурор и предложил поучаствовать в проверке организаций и предприятий на предмет использования контрафактных программ в роли эксперта по программному обеспечению 1С.

От неожиданности звонка я тут же согласился. Как-то трудно оказалось отказать прокурору, хотя я не большой любитель отвлекаться на непредвиденные дела. Потом до меня дошло, что это может быть интересно и полезно для меня. Так и оказалось.

Пока я тут крутился в компьютерно-программном мире, оказывается, в декабре 2003 года статью 146 УК РФ сделали реально работающей, и по России плавно набирает обороты машина по упаковке контрафакта и его распространителей с пользователями.

В общем, в течение недели довелось побывать с проверкой в полутора десятках предприятий и организаций – искали контрафактную продукцию 1С.

Результат проверки любопытен. В предприятиях коммерческих с этим делом все оказалось более-менее нормально, *1С-Предприятие* практически у всех проверенных оказалось лицензионное. А вот в бюджетной сфере дело швах: *1С:Бюджетная бухгалтерия* и *1С: Зарплата* и *Кадры* ворованные. Закрутились дела уголовные, и все такое. Я сейчас не об этом, а о том, что Windows у нас пока не проверяли.

Зашел я в наш компьютерный класс, прикинул стоимость ПО Microsoft, тут же раскаялся и перековался на любителя свободного ПО. А тут еще ваше дело!

Как оказалось, Open Source не только снимает проблему использования лицензионного ПО, но и жутко интересно.

АП: **Устоялось мнение, что Linux – система для профессионалов, перейти на нее без помощи специалистов сложно.**

СС: У нас в Оханске специалистов по Linux нет. Откуда? Вот мы сами по себе. Сами поставили, сами настроили, сами работаем.

В принципе, сам переход на Mandriva Linux не оказался чем-то чрезвычайно сложным, хотя шероховатости и «непонятки», конечно, были. В основном, они касались настройки видеокарт ATI (с Nvidia проблем не было) и локальной сети класса. Но со всем удалось справиться своими силами, даже не обращаясь напрямую с вопросами к сетевому сообществу. Вся нужная информация есть в Сети, и поэтому выведывать ее не пришлось. Достаточно было сделать запрос в поисковой системе.

Справились, хотя, по сути, мы обыкновенные пользователи без профильного образования.

АП: **То есть без специального образования?**

СС: Мое образование – 10 классов Оханской средней школы. Бывало, кое-где учился, но реально я обыкновенный советский «троечник». Компьютеры освоил самоучкой. Не бывал ни на каких курсах подготовки и переподготовки по этому профилю. Зачем бессмысленно тратить на это время, если есть Интернет? Технология самообучения проста и эффективна – включаю компьютер, ставлю программу и начинаю осваивать; время от времени, если что-то совсем непонятно, ищу ответы в сети.

Вот и Linux так освоил. Самое главное, что меня никто не заставлял это делать. Мне просто это интересно, и никто над душой не стоит.

АП: **А как восприняли вашу перестройку дети?**

СС: У ребят совершенно никаких проблем нет, ни испуга, ни психологического барьера. Они очень легко воспринимают новое. Больше-то проблем у взрослых.

АП: **Это вы про педагогов?**

СС: Да, мне довелось участвовать в различных дискуссиях по этому поводу. И знаете, очень удивляет позиция профессиональных учителей информатики, что перевести обучение на Linux невозможно – мол, у нас учебный план, программы, нагрузка, учебники не те. Но ведь всегда так будет! Настроение такое: нас не учили Linux, дайте денег на Windows.

АП: **Выходит, Windows рановато на свалку истории?**

СС: Я сторонник двуплатформенного обучения. Учить надо Linux, а Windows пусть остается в качестве экспоната, для факультативных занятий.

АП: **То есть, сменить акцент? А кто же обучит учителей?**

СС: Я считаю, если учитель информатики не способен самостоятельно разобраться и освоить (не торопясь, без понукания), как работать с Linux на уровне пользователя, то он просто профессионально не пригоден. Он обязан научить детей тому, как осваивать новое программное обеспечение. Это важнейший навык, который действительно стоит развивать в школе на уроках информатики. А как же это делать, если преподаватель ни разу самостоятельно не освоил серьезную программу?

АП: **Ну, по сути, программы – только инструмент; наверное, двигаться надо от целей обучения?**

СС: Что более всего меня удручает в школьном преподавании информатики, так это оторванность от реальной жизни. Учить надо так, чтобы впоследствии человек умел применить компьютер в реальной жизни так, чтобы повысить качество своей жизни.

Это мое мнение.

Я хочу научить детей, как жить в нашем «медвежьем углу» и с помощью эффективного использования Интернета не испытывать при этом комплекса неполноценности.

Компьютер – это хорошее средство от бедности. Человек, умеющий толково применить компьютер, не может быть бедным, он всегда сможет прокормить себя и свою семью.

С приходом Интернета в глубинку это можно делать в любой депрессивной территории, деревне, селе, малом городке.

Речь идет о web-издательской деятельности. Сайт – это актив, способный приносить доход, при правильном применении головы и рук, конечно.

Этому стоит учить в школе: в частности, можно и нужно не только участвовать со школьниками в разных сетевых олимпиадах, но и делать реальные проекты, оценку которым поставит не жюри из коллег, а реальная жизнь, рынок, общество.

Причем эта идея не нова – производственная деятельность в школе существует чуть ли не с древности. В советское время тоже какие-то табуретки делали. Сайты – те же табуретки, только высокотехнологичные.

Если учитель информатики – специалист своего дела, а не дутый теоретик, он может найти способ извлечь дополнительный доход из своей профессии на пользу себе, ученикам и своему образовательному учреждению.

При этом повышается значение Open Source: бесплатный инструмент повышает конкурентоспособность, наличие сообществ у таких инструментов помогает быстро наращивать квалификацию.

АП: **Вопрос традиционный: каковы дальнейшие планы?**

СС: Планов громадье! Ну, во-первых, будем дальше осваивать Linux. Во-вторых, нас занимает задача получения экономической отдачи от интернет-деятельности школьников и педагогов. А это все чрезвычайно интересно, объемно, и дел на многие годы вперед. LXF

Подходы к переходу

Сегодня **Галина Пожарина** расскажет о возможных путях миграции учреждений образования на свободное программное обеспечение.

Формирование отечественной системы образования на основе интеграции широкого спектра информационных технологий ставит перед руководителями разных уровней системные вопросы.

При проектировании системной основы использования информационных технологий и продуктов необходимо принимать во внимание их специфику. Одним из парадоксов информационных технологий можно назвать негативное влияние высокого темпа их развития.

Высокая динамика развития ИТ приводит к быстрому моральному устареванию и техники, и программного обеспечения. Для образовательных учреждений это означает необходимость учета их постоянного обновления в схеме затрат.

Очевидно, настало время принимать серьезные решения, и если приобретение лицензий на продукцию не только Microsoft, но и любого другого производителя не по карману учебному заведению, стоит серьезно изучать рынок свободного программного обеспечения. Расстановка приоритетов установки легального программного обеспечения в школе, колледже, вузе не только свидетельствует о правовом решении производственных вопросов, но и (а об этом нельзя не думать именно в образовательном учреждении) формирует из учащихся грамотных в правовом отношении специалистов.

Следует отметить, что и государственные образовательные стандарты, и примерные учебные программы , которые являются ключевым элементом российской системы образования, не дают прямой адресации к конкретным программным продуктам конкретных производителей. Прямые указания на использование конкретной операционной системы в перечне программных средств, необходимых для реализации образовательных стандартов, также отсутствуют.

Такая позиция позволяет руководителям образовательных учреждений и службам технической поддержки принимать во внимание эффективность внедрения свободных программных решений при решении вопроса о комплектации компьютерных лабораторий оборудованием и программным обеспечением. Пути миграции на свободное ПО могут быть различными и будут зависеть, в первую очередь, от перспективы развития технической базы образовательного учреждения.

1 Путь первый: «Новый хард и свободный софт».

При возможности приобретения современного оборудования общего назначения (исключаем узкопрофессионально-ориентированное оборудование – такое как комплексы для программируемых станков с поставляемым в комплекте программным обеспечением – в данном случае профессиональная направленность оборудования обусловит использование специализированного ПО, под технические характеристики которого будет сконфигурирована операционная система) резонным будет стремление установить современное программное обеспечение (последнего года выпуска). Причина – оборудование предполагается к использованию в течение примерно пяти лет (стандартный срок полной амортизации компьютерной техники).

При внедрении свободного программного обеспечения при данном подходе достигаются следующие цели:

- с позиций экономии свободное лицензионное программное обеспечение на порядок дешевле конкурентных несвободных программных решений;
- в правовом аспекте использование свободной лицензии позволяет распространить программные продукты в качестве базовых учебных материалов без дополнительных значимых затрат со стороны учащихся и их родителей (социальный аспект) без провокаций к приобретению пиратских копий программных продуктов.
- функциональные возможности современного программного обеспечения отвечают актуальным запросам пользователей (богатая графика, мультимедийные компоненты, устойчивость к вирусным атакам, интегрированные среды, наличие виджетов и пр.), в первую очередь подростковой аудитории как наиболее требовательной к ресурсам.

2 Путь второй: «Новый сервер, свободный софт».

Приобретение нового оборудования (в первую очередь, рабочих станций) затруднительно.

Возможна интеграция в систему одного мощного сервера.

Реализация решения такова:

- имеющиеся в наличии рабочие станции будут выполнять роль терминалов;
- на сервер устанавливается современное программное обеспечение.

Эти два пути предполагают:

- интеграцию операционной системы
- подбор пакета программного обеспечения, соответствующего решаемым учебным задачам.

Учитывая принцип сборки дистрибутивов Linux, входящий в их состав набор программ позволяет не только реализовать диалог пользователя и компьютера (операционная система), но и решать прикладные задачи, поэтому доустановка дополнительного программного обеспечения может и не потребоваться.

3 Путь третий «Свободный софт на несовременном железе».

Допустим, оборудование не планируется обновлять вообще. Тогда имеющееся аппаратное обеспечение при исследовании актуальности лицензий на установленный софт может быть перепрофилировано на свободное ПО, как полностью, так и частично.

- Полное профилирование (операционная система и пакет прикладных программ и утилит) требует подбора пакета, соответствующего мощности используемой компьютерной техники. Учитывая возможные сбои в работе техники (в силу естественных причин ее старения), к пакету предъявляются требования повышенной устойчивости к сбоям, нетребовательности к ресурсам и пр. Повышаются и квалификационные требования к персоналу, обслуживающему подобные технические решения.
- Частичное профилирование (установка прикладных программ и утилит на оборудование с работающей операционной системой – скорее всего, Windows) – наиболее щадящий путь при использовании кроссплатформенных решений. При частичном профилировании на свободное программное обеспечение снижается нагрузка на пользователя за счет ограниченного внедрения неизвестного программного обеспе-

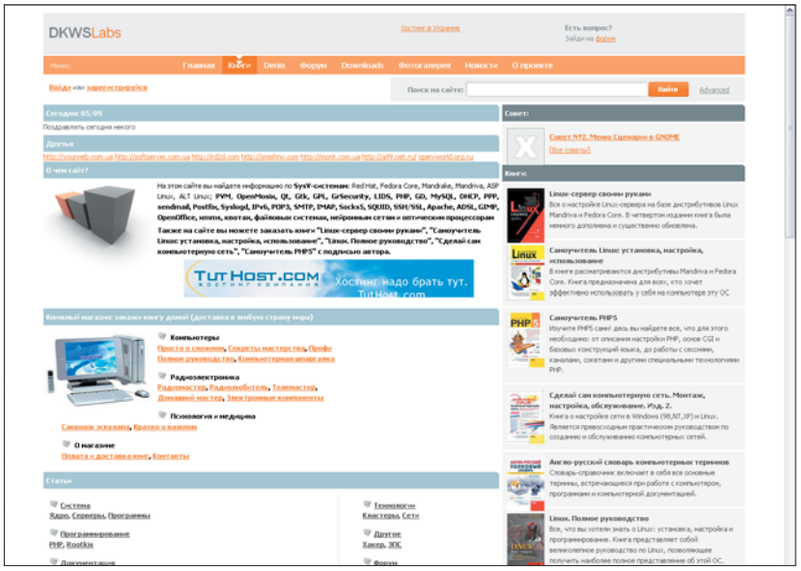
чения в короткий отрезок времени. Такой принцип миграции позволяет поэтапно вводить в активное пользование программное обеспечение, руководствуясь его значимостью для пользователя.

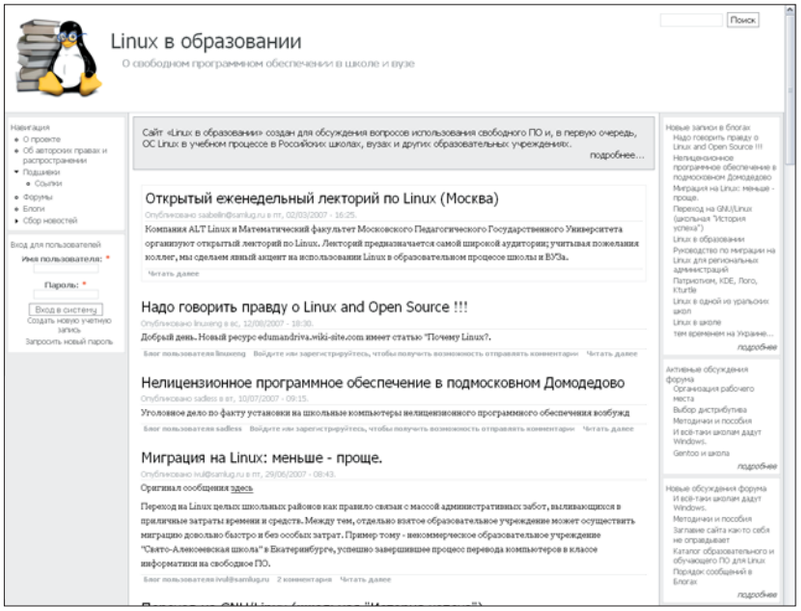
Учет вышеперечисленных аспектов делает выбор программного обеспечения достаточно дифференцированным не только на региональном уровне, но и на уровне отдельных учебных заведений. Сопровождение технической документацией и специальной литературой может быть реализовано с применением профильных ресурсов Интернета.

» Материалы виртуальной энциклопедии «Linux по-русски» (**http://www.rus-linux.net**) позволяют не только сориентироваться в достаточно обширной линейке программных продуктов, но и принять определенные решения в их выборе для конкретных инсталляций.

» Опыт практиков перевода образовательных учреждений на свободное ПО можно изучить, анализируя материалы блогов и форумов ресурса Linux в образовании (**http://linuxed.ru**).

» На авторском сайте Дениса Колисниченко (**http://dkws.org.ua**) размещена богатая подборка материалов по Red Hat, Fedora Core, Mandrake, Mandriva, ASP Linux, ALT Linux. Для практиков будут полезны издания «Linux-сервер своими руками», «Самоучитель Linux: установка, настройка, использование», «Linux. Полное руководство», «Сделай сам компьютерную сеть» и др. LXF



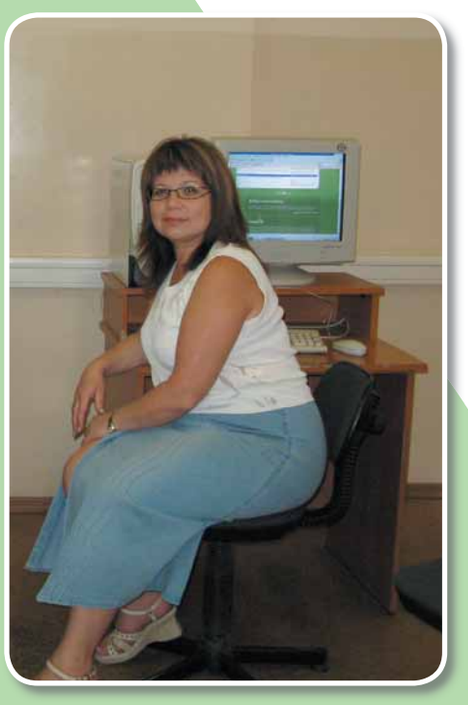


Западный форпост России

Сегодня мы совершим виртуальное путешествие в самую западную область России – город Калининград, чтобы посетить лицей № 22, который вот уже более года использует Linux. **Евгений Балдин** вступил в переписку с его мастером производственного обучения – **Ириной Алаевной Красильниковой** и **Артемом Сергеевичем Золочевским** – инструктором по GNU/Linux в учебном центре РОСТ. И вот что они ему рассказали...

Ирина Алаевна Красильникова

Школы выбирают GNU/Linux в том числе потому, что он бесплатен. Но встает вполне резонный вопрос: «А во сколько обойдется переобучение преподавательского состава?» Не проще ли оставить все как есть, затянуть пояса и прикупить привычные программы? Неприятно, что эта «боязнь новых знаний» возникает в школах – центрах обучения и просвещения. И не так страшен зверь переобучения, как его малюют. В поддержку этого тезиса выступит наше следующее интервью.



Сегодня Евгений Балдин беседует с **Ириной Алаевной Красильниковой** – мастером производственного обучения в профессиональном лицее № 22 в г. Калининграде.

Linux Format: Лицей № 22 чем-нибудь отличается от других школ вашего города? Какова специализация вашего лицея?

Ирина Алаевна Красильникова: Во-первых, это учреждение профессионального образования, где учащиеся вместе со средним образованием получают профессию. Надо сказать, что лицей по профилю не связан с IT-индустрией. Мы готовим продавцов, коммерсантов, маркетологов и бухгалтеров. Но на первой ступени подготовки бухгалтеров наша программа предполагает получение профессии «оператор ЭВМ».

LXF: Какова причина перехода на свободное ПО?

ИАК: Ну, начиналось все с коммерческих вопросов. А как познакомилась, то заинтересовала и техническая сторона. Для меня было открытием, что с Linux можно изучать (как бы изнутри) как устроено программное обеспечение, а не просто обучать пользованию им. Правда, эту мысль пока трудно донести до детей. Думаю, отчасти это связано с тем, что все же это для них непрофильные вещи. Хотя с другой стороны, даже если ограничиваться обучением тому, как пользоваться компьютером, то «какая разница»? Там кнопки – тут кнопки. Там меню – тут меню. «А если не видно разницы – то зачем тратить…» [улыбается]

LXF: Каково «наполнение» кабинета информатики?

ИАК: Все машины у нас практически одинаковые: Celeron 800, ОЗУ 128 МБ, интегрированное видео, жесткие диски по 10–40 ГБ и 15-дюймовые мониторы. У нас также есть общий принтер для печати. К машине преподавателя подключен мультимедиа-проектор, который позволяет показывать просто на белой стене класса изображение с машины преподавателя. Это очень удобно.

LXF: Расскажите о процессе перехода поподробнее…

ИАК: Предыстория такая. У нас два компьютерных класса: по десять и пятнадцать рабочих мест. Первый был получен по Президентской программе и был полностью укомплектован лицензионным ПО (MS Windows). А вот со вторым весной 2006 года назрела необходимость решать вопрос используемого ПО. Надо сказать, что это было еще до «дела Поносова». Так что можно сказать, что мы пришли к этому сами.

Все началось с разговора с бухгалтером о суммах, которые придется вложить в комплектование второго компьютерного класса легальным ПО. И тут родилась первая идея, которая приходит на ум, наверное, всем, кто хочет потратить как можно меньше денег: «А есть ли альтернативные пути?» На тот момент я ничего не знала о существовании альтернатив, хотя впоследствии, когда уже все было сделано, я вспомнила, что уже что-то слышала об этом.

Мы обратились к знакомому бухгалтеру с вопросом о том, есть ли варианты, и он рассказал кратко об альтернативах в виде GNU/Linux и дал телефонный номер и адрес Учебно-консультационного центра «РОСТ», где, по его сведениям, открылись курсы по GNU/Linux. Созвонившись, я записалась на курс для новичков, и приобрела там первый свой дистрибутив – ALT Linux Master 2.4. После прохождения курса, а вернее, уже даже во время обучения, стало понятно, что GNU/Linux может быть не только выгоден с коммерческой точки зрения. Он вполне может решать учебные задачи в нашем лицее. В классе для операторов ЭВМ был установлен этот дистрибутив, и началась работа по адаптации учебных планов к новым условиям. Надо сказать, что переход был сделан практически моментально, без подготовки. Прямо в середине учебного года. И пошла нормальная работа – уроки, зачеты, аттестации.

LXF: Насколько сложно все было организовать? Сколько это заняло времени?

ИАК: После окончания курса я самостоятельно со старшекурсниками установила ALT Linux Master 2.4. На это ушло примерно 3 дня. Начали заниматься. Поддержку техническую и теоретическую мне оказывает Артем Золочевский – это мой преподаватель на курсах, которые я прошла. Он же позже посоветовал обновить дистрибутив до ALT Linux Compact 3.0, что и было им сделано в начале февраля 2007 г. совместно с нашими студентами. Сейчас готовимся тоже обновляться. Ждем, пока выйдет ALT Linux 4.0.

А что было сложным, так это «охи» студентов, ну и нужно было откорректировать учебные программы. Но во время работы с удивлением убедилась, что менять пришлось не так и много. Только практические примеры. Что еще раз подтверждает то, что использование ПО построено на универсальных принципах, и чтобы показать принципы работы на компьютере, совсем не обязательно тратить большие средства. Ведь, по сути, все везде одинаково – кнопки, меню, окна.

Основные «претензии» – непривычно, непонятно. Еще на быстродействие жалуются – техника у нас не самая новая. Хотя знаете, сейчас получили MacBook'и – оказывается, там тоже внутри Unix. Еще не смотрели на них внимательно. Так что даже интересно. Там на них предустановлены Windows и MacOS. Теперь эта новизна меня даже не пугает, даже интересно познакомиться еще с одной системой. В Windows будем однозначно ставить свободное ПО: *OpenOffice*, *Inkscape*, *GIMP* и прочее.

LXF: Как ко всей вашей затее носится администрация?

ИАК: Администрация меня всецело поддержала. Даже оплатили курсы. Никаких проблем тут просто нет. Директор в курсе и полностью доверяет, да еще и средства экономит. Разве это может ее не радовать?

LXF: Кто-нибудь из детей перешел на GNU/Linux? Появились ли заинтересованные?

ИАК: Да, были такие. Сказали – возьмут дистрибутив. Но гораздо больше берут свободного ПО под Windows. К примеру, *OpenOffice* очень часто спрашивают. Главное, чтоб дети поняли, что есть выбор. И, кажется, они это поняли.

LXF: Если сравнить ситуацию до и после, в чем основное отличие?

ИАК: Для начала «о земном»: чувствую себя абсолютно спокойно – не страшны всякие там проверки и прочее. Ну и просто интересно. Однозначно, расширила свой кругозор. Теперь мир не сходится на какой-то определенной программе: оказывается, есть выбор! Надо просто о нем знать.

LXF: Ваше образовательное учреждение уникально в пределах Калининграда, в смысле перехода на свободное программное обеспечение?

ИАК: Насколько мне известно, в сфере профессионального образования мы единственные. А вот насчет обычных школ – по-моему, есть еще одна, где внедрен GNU/Linux, № 12. Но это было уже после нас, так что мы первые, хоть и не единственные.

LXF: Какие планы на будущее?

ИАК: Продолжать работу в GNU/Linux. Этот класс у нас обречен на использование GNU/Linux [*улыбается*]. Ну и будем программы далее переделывать. Хотелось бы, чтоб мы не были «белыми воронами» и нашлись последователи и в системе профессионального образования. Уже нашлись заинтересованные, взяли координаты. Пока в мыслях есть организовать встречу для моих коллег из профессиональных лицеев и показать им, с Артемом, что у нас получилось.

LXF: Чего бы вы пожелали бы сообществу FOSS и GNU/Linux – в частности, для облегчения процесса внедрения в школы?

ИАК: Тут много чего можно сказать. Вот бухгалтерию бы мне под GNU/Linux. Что-то типа «Консультанта». Я хоть и знаю, что это возможно запускать в *Wine*, но еще не пробовала. Есть проблема у меня и с *OpenOffice.Base*. В той версии, что у меня сейчас, не работают отчеты, и база не открывается на другой машине. Надеюсь, в следующих версиях это будет исправлено. С флоппи-приводами как-то надо, чтоб удобно было работать. Да много можно и нужно улучшений делать. Документации побольше хорошей, и на русском языке – это и для меня, и для детей большая помощь. Но это скорее работа для «технических специалистов». Для меня как преподавателя, наверное, еще важнее иметь поле для общения, обмена опытом, учебными планами, где можно спросить, получить помощь от таких же, как и ты, «практиков» преподавания.

Артем Сергеевич Золочевский

Научить учителя – отнюдь не простая задача, но **Артему Золочевскому** (artem.zolochevskiy@gmail.com) это определенно удалось. Артем ведет платные курсы GNU/Linux в Учебно-консультационном центре РОСТ. Вся его профессиональная деятельность так или иначе связана со свободным программным обеспечением.

LXF: Насколько востребованы платные курсы GNU/Linux?

Артем Сергеевич Золочевский: Они вполне могут быть востребованными. Стоимость курса чуть меньше стоимости аналогичных Windows-курсов в нашем городе. Пока мы стараемся придерживаться такой политики. Курс должен быть доступен.

Когда мы начинали, я, честно говоря, достаточно скептически относился к идее организации подобного обучения. Но на сегодняшний день вижу, что настало время «собирать камни». Люди приходят, интересуются. Да и просто в разговорах: если раньше я произносил слово «Linux», а в ответ, как правило, тишина, то сейчас большинство собеседников, особенно молодые люди, это «странное» слово уже где-то слышали. Так что интерес, думаю, будет расти, а значит, надо предлагать и образовательные услуги.

LXF: Вы являетесь одним из разработчиков и сотрудничаете с ALT Linux. Что вы можете сказать о дистрибутиве этой компании применительно к школе?

АСЗ: Я состою в ALT Linux Team. Это команда людей, участвующих в разработке Sisyphus, на основе которого создаются дистрибутивы ALT Linux. Сейчас «горячая пора». Готовится целая линейка дистрибутивов, где будет и специализированный «школьный» дистрибутив JuNIOr 4. Он имеет относительно небольшие требования к оборудованию: основное – это 128 МБ памяти, ну, и процессор класса Pentium III. Состав дистрибутива с избытком покрывает школьную программу по информатике и имеет значительный запас программ на DVD. Инсталлируется с минимальным участием пользователя – в основном при решении, куда ставить, и для настройки сети. Есть поддержка терминальных возможностей, так что при желании (и, возможно, помощи специалиста) старые/слабые машинки могут использоваться как терминалы. Надеюсь, такой дистрибутив снимет часть «груза» по внедрению GNU/Linux в образовательных учреждениях.

LXF: Чего с вашей точки зрения не хватает для упрощения внедрения GNU/Linux в школе?

АСЗ: Преподавателю не хватает хорошей документации на русском. Вот специально прошелся по книжным магазинам. Да, ситуация явно лучше, чем пару лет назад. Но это все равно капля в море. Так что надо помочь преподавателям хорошей литературой.

Ну и важно предоставить преподавателям-первопроходцам поле для общения, обмена опытом, материалами. Преподаватели, взявшиеся за GNU/Linux, и так уже огромные шаги сделали. Они адаптируют свои учебные планы, занимаются самообразованием. Нам нужно просто немного помочь им. Это что касается непосредственно преподавания. Ну и есть проблемы технического характера. Вот даже банально, где взять преподавателю дистрибутив? С кем посоветоваться? Ведь даже LUG не в каждом городе есть. Очень важно иметь кого-то, кто сможет помочь с решением технических проблем: как установить что-то, организовать сеть, да просто советом помочь.

Пара слов от студентов

Артем Золочевский (Артем) подзывает молодого человека…

Артем: **Ну, что скажешь о Linux?**

Алексей Вишневский: [немного недовольно] Работать без знания командной строки нереально.

Артем: **И это минус?**

Алексей: [уже более дружелюбно] Да нет, просто знать все надо.

Артем решил опросить и представительницу прекрасного пола…

Артем: **А ты что скажешь?**

Алина Рожкова: Да нормально, мне даже больше нравится. Красивая графическая оболочка.

Артем: **А чем вы в нем (GNU/Linux) занимаетесь-то?**

Алина: Мне нравится, что можно безболезненно ходить по Интернету [улыбается]. Еще нам рассказывали про командную оболочку *Bash*, мы работали в *mc*, смотрели, как устроена *X*.

Артем: **А что сейчас изучаете?**

Алина: А сейчас работаем в *OpenOffice*. Создаем в нем листовки о лицее для приемной комиссии. Это у нас творческое задание такое. LXF

Mandriva FLASH

Mandriva Linux на 2GB USB накопителе

3D десктоп у Вас в кармане!

Mandriva Flash - это предустановленный дистрибутив Mandriva Linux на 2GB USB накопителе. Возьмите с собой накопитель, пусть Linux будет всегда с Вами, храните и обменивайтесь данными на целом гигабайте свободного пространства! Это не только практично, просто и удобно, но и высокопроизводительно, и современно.
Mandriva Flash представит Вам мир, в котором размер не имеет значения!

Flash - новейший продукт разработки Mandriva!

Испытайте реальный Linux-десктоп, храните документы и настройки на накопителе - не нужно ничего устанавливать на винчестер.
Удивите друзей мобильным 3D-десктопом.
Наслаждайтесь новейшими 3D-технологиями в дороге, покажите друзьям все, что может Linux на их машинах.

Подарите Linux любимому человеку !

Mandriva Flash - превосходный подарок для любого из Ваших друзей, незнакомых с Linux, или для Вашего спутника жизни!

Mandriva Flash - компактный и надежный высококачественный USB-накопитель. При 5-летней гарантии известного производителя Вы можете не бояться за свои частные данные.

Закажите Mandriva FLASH на сайте: www.linuxcenter.ru

Список магазинов партнеров: www.linuxcenter.ru/mag.phtml

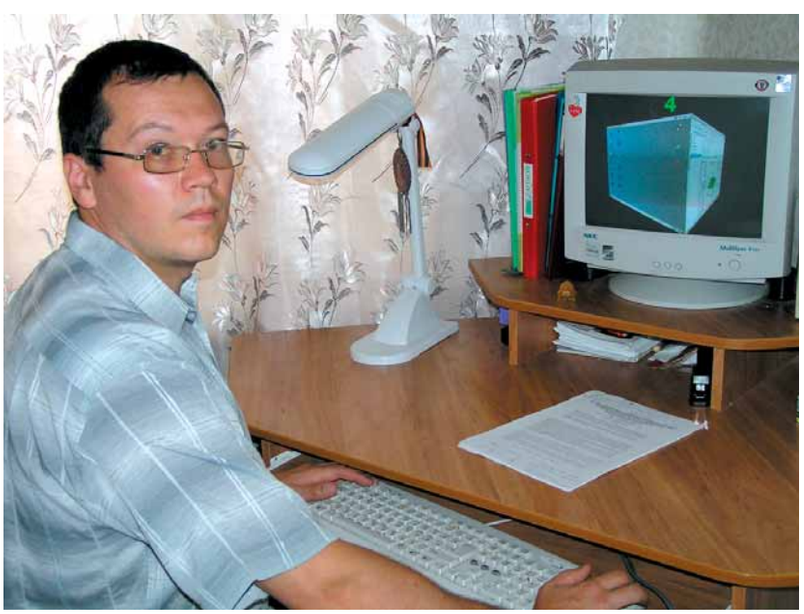
Авторизованные курсы Mandriva Linux: www.linuxcenter.ru/training

Mandriva

Linux center
www.linuxcenter.ru

Российские информатики физмата выбирают GNU/Linux

Это событие произошло на родине Ивана Сергеевича Тургенева – в городе Орле. Кафедра информатики физико-математического факультета Орловского государственного университета действительно сделала выбор в пользу GNU/Linux. Это было не просто, и именно об этом **Илья Евгеньевич Ермаков**, зам. директора ООО ОЦПИ «Метасистемы», будет говорить с заведующим кафедры информатики, к.ф.-м.н. доцентом **Дмитрием Николаевичем Никольским.**



ИЕЕ: **Дмитрий Николаевич, расскажите, пожалуйста, что послужило поводом для совершившегося полного перевода учебного процесса на открытое ПО?**

ДНН: Можно сказать, что повод тривиален – мы, как и миллионы наших соотечественников, в начале 2007 г. стали свидетелями всем известного судебного процесса.

ИЕЕ: **Можете вспомнить, какие у вас и ваших коллег были впечатления от тех событий?**

ДНН: Если оставить в стороне абсурдность показательных судов над школьными учителями, то в стране начались вполне предсказуемые процессы – осознание того, что за коммерческие продукты зарубежных корпораций нужно платить, и платить много, гораздо больше того, чего они в реальности стоят. Многолетняя «халява» была огромным тормозом для распространения открытых систем и развития отечественной ИТ-индустрии. Несмотря на трудности, которые будут сопровождать период перехода и адаптации, процесс, видимо, уже пошел – и мы это только приветствуем.

ИЕЕ: **Какая ситуация с ПО и его легальностью была в ОГУ до начала перехода на открытое программное обеспечение?**

ДНН: В учебном процессе на специальности «Прикладная математика и информатика» уже пять лет использовался Linux и GNU. Но значительная часть ПО продолжала работать под Windows – на нем учили будущих учителей, на нем преподавали информатику для непрофильных специальностей других факультетов. Все Windows-ПО в учебных аудиториях Вычислительного центра (ВЦ) было нелицензионное. Наша кафедра посчитала оптимальным решением полный переход на GNU/Linux.

ИЕЕ: **Насколько велик парк машин в ВЦ ОГУ?**

ДНН: Парк из приблизительно 60 машин (5 классов, Intel Celeron 2 ГГц, 256 МБ) и 3 серверов (2 работают под Red Hat 9 и один под ASP Linux 10, а сейчас 11.2). Все Windows-ПО на всех машинах, за исключением серверов, было установлено сотрудниками ВЦ ОГУ для обеспечения учебного процесса. Сервера настраивались сотрудниками нашей кафедры, и на них уже 4 года работает достаточно сложная связка Samba, LDAP и NFS, обеспечивающая всю эту гетерогенную сеть едиными каталогами и профилями пользователей для Windows и Linux. Ранее сервера GNU/Linux обеспечивали также управление доменом Windows, сейчас это не делается, ввиду отсутствия последнего.

ИЕЕ: **Когда было принято решение о переходе на открытое ПО и как оно претворялось в жизнь?**

ДНН: В феврале, до начала учебного семестра, кафедра приняла решение прекратить использование нелегального ПО. Первым делом был уничтожен раздел Windows на кафедральных машинах и в собственных классах физмата. Нашему примеру последовали сотрудники ВЦ, удалив нелегальную Windows в оставшихся классах. Вышедшие на занятия студенты встретились уже непосредственно с GNU/Linux.

ИЕЕ: Какой дистрибутив был выбран в качестве замены Windows?
ДНН: Для рабочих станций был выбран дистрибутив ASP Linux 11.2, поскольку предыдущие версии этого дистрибутива уже ранее использовались на кафедре и имелся опыт по их настройке. Кроме того, дистрибутив имеет прекрасную русификацию и неплохую настройку «из коробки», что важно при большом парке машин.

ИЕЕ: Много сложностей пришлось преодолеть?
ДНН: Сложности, конечно, были. Студентам учительских специальностей, а также студентам не физико-математического факультета пришлось адаптироваться к новому ПО. Следует отметить, что процесс адаптации прошел быстро, без каких-либо нарушений учебного процесса. Переход на GNU/Linux вызвал дополнительную нагрузку на преподавателей. Им пришлось ознакомиться с новым инструментарием и внести соответствующие корректировки в учебные планы. Наиболее активным преподавателям пришлось найти время для установки и настройки нового программного обеспечения. Несмотря на все трудности, с поставленной задачей преподаватели успешно справились, взяв на себя все техническое обеспечение учебного процесса на базе открытого ПО.

ИЕЕ: Поддержал ли переход на GNU/Linux обслуживающий персонал ВЦ?
ДНН: К сожалению, текущий руководитель ВЦ не поддержал идеи перехода на свободное программное обеспечение, и не поддерживает ее и в настоящее время. Сервера и рабочие станции, работающие под управлением GNU/Linux, были установлены и поддерживаются только сотрудниками нашей кафедры.

ИЕЕ: Дмитрий Николаевич, какие программные продукты использовались ранее в учебном процессе и чем они были заменены при переходе на GNU/Linux?
ДНН: Если говорить о пользовательских курсах, то, например, *Microsoft Office* был заменен на *OpenOffice*, *Adobe Photoshop* – на *GIMP*, *3DStudio Max* – на *Blender*... Как показал опыт, *OpenOffice 2.1* в каких-то аспектах более удобен, а в каких-то требует доработки, но, безусловно, он пригоден для замены аналогичного продукта от Microsoft. О приложениях *GIMP* и *Blender* имеются только положительные отзывы. С чем вообще не возникло проблем – так это с системой компьютерной верстки *LaTeX*, которая изучается у нас на всех специальностях физмата. Вместо *MiKTeX* стали использовать входящий в поставку ASP Linux *teTeX*, вместо редактора *TeXShell* – несравненно более удобный *Kile*, вместо векторного редактора *TeXCAD* – на порядок более мощный *Xfig*, появилась возможность пользоваться редактором диаграмм *Dia*.

ИЕЕ: Вы не сказали ничего про инструментарий программирования...
ДНН: Так ничего из этой категории ПО заменять и не потребовалось. Вот уже пять лет мы учим системных программистов исключительно на базе Linux и соответствующих систем программирования. Студенты работают с компиляторами GCC, средой *KDevelop*. Системное программирование изучают на базе стандартов POSIX. Глубоко осваивают графическую библиотеку *Qt*, которая является кроссплатформенной. Изучают технологии распределенных вычислений на базе интерфейса MPI с дистрибутивом MPICH. На учительских специальностях заменой для *Turbo Pascal* и Delphi послужили *Free Pascal* и *Lazarus*.

ИЕЕ: Это прекрасно. Однако сегодня часто можно слышать возражения, что выпускники ВУЗа с такими знаниями могут оказаться невостребованными на массовом рынке труда, который плотно сросся с Windows-системами...
ДНН: В этом вопросе все зависит от видения задач высшего образования. Роль ВУЗа нельзя опускать до уровня профтехучилища – до обучения пользованию конкретными технологиями, то есть до подготовки узких специалистов, способных решать типовые задачи типовым инструментом.

Все становится на свои места, если преподавать не конкретные системы, а общие принципы программной инженерии. Стандартные курсы ориентированы на постановку правильного и широкого мышления у студентов. Например, курс «*Языки программирования и методы трансляции*» ориентирован на изучение внутреннего устройства трансляторов, а не какой-то конкретной IDE для их разработки. Наиболее удобным средством для отработки на практике теоретических знаний, полученных в этом курсе, оказались генераторы *flex* и *bison*, входящие в базовую конфигурацию GNU/Linux. Курс «Базы данных и экспертные системы», в котором используется СУБД *MySQL*, позволяющая, в отличие от распространенных в образовании *Access* и *FoxPro*, показать в чистом виде ключевые идеи реляционных БД, научить студентов с нуля проектировать и реализовывать приложения баз данных, показать все основные звенья этих приложений, не скрытые за окошками и кнопками. Курсы *«Операционные системы»*, *«Программирование на языках высокого уровня»*, в практикуме которых используются высокоуровневые языки системного программирования Ada (пакет *AdaCore GNAT*) и *Oberon* (среда *BlackBox Component Builder*), вырабатывают навыки применения в разработке высокоуровневых подходов, абстрагирования от особенностей конкретной платформы. Сегодня это умение становится особенно ценным – с демонополизацией рынка ОС будут становиться все более востребованными высокоуровневые кроссплатформенные технологии. Переход на GNU/Linux оказался удачным для курса *«Программирование на ассемблере»* – работа с ассемблером под Linux оказалась гораздо более удобной, чем под Windows, не только из-за его синтаксиса, но и из-за возможности напрямую обращаться к стандартной C-библиотеке. Спецкурсы, связанные с программированием для Интернета, преподаются на базе Linux наиболее естественным образом.

ИЕЕ: Так все же, как вы считаете, удастся ли порвать порочный круг, сложившийся в нашей отрасли?
ДНН: Трудно предсказывать будущее. Но можно точно сказать, что разорвать этот круг можно только в высшем образовании. Как в индустрию, так и в школы должны прийти специалисты, готовые использовать все разнообразие открытых технологий, а не зацикленные на единственной импортной системе. И мы в Орловском регионе сделали первый шаг к этому. Через год-два в школы придут учителя, а в бизнес – программисты, которые в ВУЗе работали исключительно с Linux. А это значит, что уже очень скоро Windows и Visual Studio можно будет называть «альтернативными системами»...

Прим. редакции: На официальном сайте ОГУ **http://www.univ-orel.ru/** все документы представлены в формате Microsoft Word. LXF

Справится ли пингвин со школьной нагрузкой?

Говорят, что в ближайшее время в окна кабинетов информатики учебных заведений может постучаться пингвин. **Светлана Шапошникова** разбирается, стоит ли впускать его внутрь.

Наш эксперт

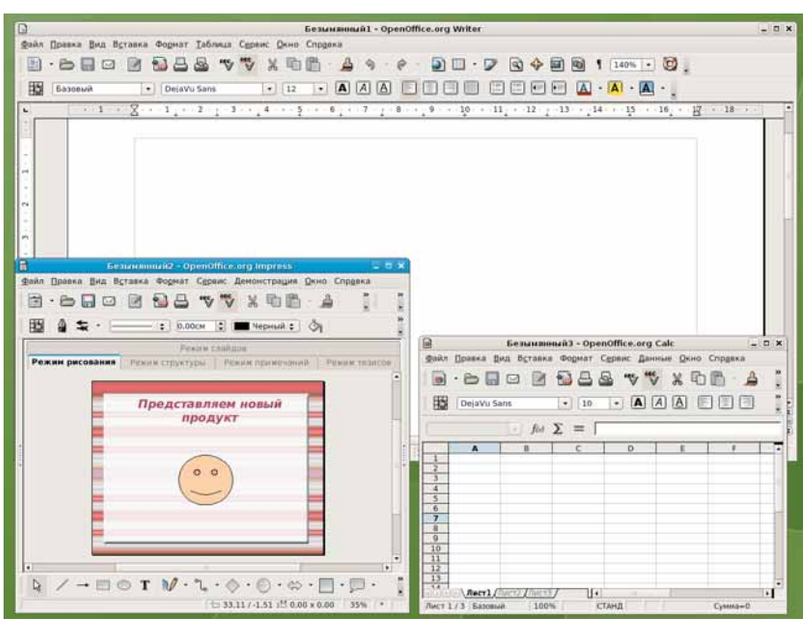
Светлана Шапошникова
Педагог дополнительного образования кружков компьютерной техники ЦД(Ю)ТТ «Городской» г. Липецка, использует Linux в образовательном процессе с 2005 года.

Говорят... А стоит ли верить и готовиться «махать ручкой» уже почти родной Windows? Согласитесь, что перспективы того или иного продукта зависят от его возможностей, а также удовлетворения им текущих потребностей. Оценим, насколько удовлетворяет Linux запросам образовательных учреждений для юных граждан. Сначала выясним, имеет ли «пингвин» необходимый минимум программного обеспечения, а затем посмотрим, подходит ли он для углубленного и какого бы там ни было изучения компьютерных технологий.

Уровень «Стандарт»

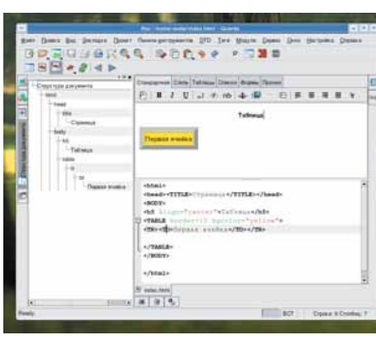
Учебные программы для школ обычно регламентируют, что должно изучаться и какими знаниями, умениями и навыками должен владеть ученик в итоге. Если Linux не содержит необходимого минимума приложений для освоения учебной программы, то его шансы оказаться в школе будут близиться к нулю. Взглянув на рекомендуемые программы и пролистав несколько школьных учебников по информатике, делаем вывод о минимуме необходимого ПО. Итак, обязательно должны быть: клавиатурный тренажер, редакторы текстовой, графической и числовой информации, приложение для создания презентаций, система создания баз данных, средства для разработки web-сайтов и система программирования.

Клавиатурный тренажер в Linux имеется: входит в состав пакета *kdeedu* и называется *KTouch*. Включает упражнения на множестве языков, а в последних версиях и на русском тоже. Правда, русской раскладки обнаружить не удалось, наверное, ее еще не успели сделать...



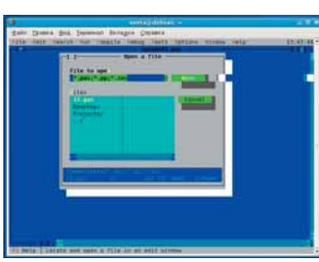
Свободный офисный пакет *OpenOffice* с радостью займет почетное место «инструмента первой необходимости», которое пока еще принадлежит *Microsoft Office*. *Writer*, *Impress*, *Calc* и *Base* вряд ли чем уступят *Word*'у, *PowerPoint*'у, *Excel*'ю и *Access*'у соответственно.

Что касается графических редакторов, то и в них недостатка нет. *KoulorPaint* – почти тот же *Paint*, но куда приятней в работе. *Draw* из все того же открытого офиса может послужить неплохим редактором векторной графики, а *GIMP* прекрасно заменит *Photoshop*.

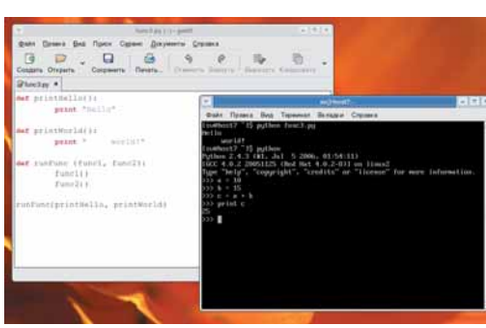


Инструментов для создания HTML-страниц тоже предостаточно. Особо хотелось бы выделить *Quanta Plus*: после перехода с *MS FrontPage* особой разницы и головной боли замечено не было.

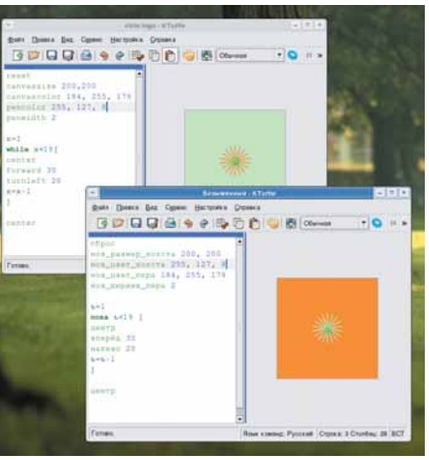
И, наконец, должен иметься транслятор пригодного для обучения языка программирования. Возможно, для этих целей подойдет интерпретируемый язык Python. Из положительных сторон Python'а можно отметить следующее: структура кода визуально воспринимается легко; интерпретатор имеется почти в любом дистрибутиве Linux; можно работать как в режиме интерпретатора, так и запускать программу на выполнение после написания кода. Однако все желающие хранить верность Pascal'ю смогут найти в Linux компилятор *Free Pascal*, представляющий собой его аналог.



Напоследок следует выяснить, имеется ли для Linux англо-русский словарь. Хотя он и не является таким уж принципиальным компонентом кабинета информатики, но часто используется в образовательном процессе. На глаза попался *StarDict*, при ближайшем рассмотрении – неплохой заменитель *Lingvo*.



Ну вот, пожалуй, и все: пингвин благополучно освоил образовательный минимум. Попробуем двигаться дальше.



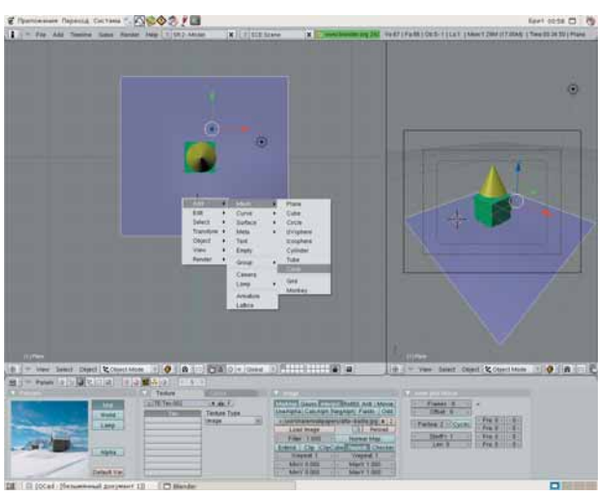
Уровень «Продвинутый»

Если раньше предмет «Информатика» был прерогативой старших классов, то на сегодняшний день его изучение нередко начинается намного раньше, иногда даже в младших классах. Кроме того, появилось множество УМК (учебно-методических комплексов) по информатике для классов с углубленным изучением того или иного предмета (так называемых «профильных классов»). Все это требует дополнительных программных средств.

Обычно классе в пятом, помимо дежурного *Paint*'а, на практических занятиях изучаются еще и лого-миры. В Linux также имеется среда для изучения языка Лого – *KTurtle* из пакета *kdeedu*. В последних версиях «линукс-черепахи» появилась возможность устанавливать русский язык команд (хотя с необходимостью этого можно и поспорить).

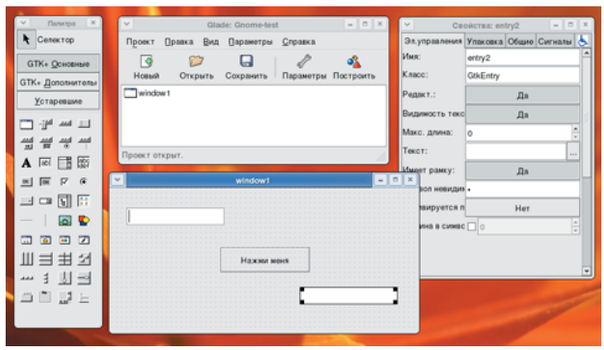
Для старших классов в первую очередь потребуются среды программирования наподобие *VisualBasic* и *Delphi*. Конечно, существует *Lazarus* – свободный аналог Delphi, но можно попробовать и «родные» Linux'у системы разработки. Правда, это не совсем «системы»: обычно создание графического интерфейса и написание кода происходят в разных приложениях. Поначалу это может испугать «разработчиков» методической литературы, но, с другой стороны, данный способ «раздельного питания» начинающих программистов может дать более глубокое понимание этапов создания программы.

Кроме того, нередко в среде Windows изучаются *AutoCAD* и *3ds Max*. В Linux их могут заменить *Qcad* и *Blender*.

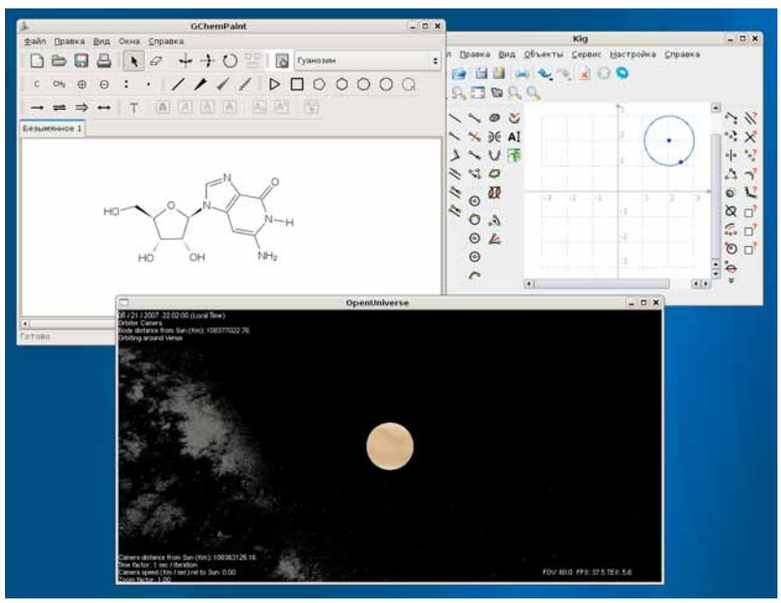


А что для профильных классов? Математика, химия, физика... Может ли Linux удовлетворить их запросы? Вот что удалось найти, глубоко не копаясь в программном обеспечении для различных сфер. *Kivio* – редактор схем из пакета *KOffice* – аналог *MS Visio*: содержит обширную базу объектов, в том числе и по физике (электричество и т.п.). Для математиков – *Kig* (интерактивная геометрия), *Maxima* (алгебраическая система) и еще множество приложений. *KStars* – настольный планетарий – может оказаться хорошим дидактическим материалом при изучении астрономии, но воображение поразил некий *OpenUniverse Space Simulator*, позволяющий виртуально путешествовать в космическом пространстве [следует также обратить внимание на *Celestia*, – прим.ред.]. Биолого-химическое направление тоже не останется в обиде: нашлись приложения как для двумерного, так и трехмерного редактирования формул молекул.

Свои требования к «пингвину» может предъявить и «объект обучения», и среди них, следует подозревать, будут игры. Что на это скажет Linux? Возможно, здесь и нет той игровой «крутизны», что в Windows, но уж многообразие обеспечено. К тому же, к услугам ученика предоставляется удобный переключатель рабочих столов (особенно в Gnome), призванный легким движением мыши делать незаметными любые шалости.



У Linux много преимуществ, и вряд ли он сможет долго скромничать по этому поводу. И теперь, отвечая на серьезнейший вопрос «справится ли пингвин с нагрузкой, возлагаемой на него предметом информатика в стенах российских школ?», следует сказать «да».



Тут немного другая проблема: справятся ли с пингвином учителя? LXF

LINUX FORMAT

В помощь FREE новичку

Сегодня мы открываем новую рубрику, посвященную разработке компьютерных игр. Слово для приветствия предоставляется идеологу Игростроя LXF – **Андрею Прахову**.

Наш эксперт

Андрей Прахов
Вращается в сферах СМИ, активно продвигая идею Open Source в своем окружении. Участник нескольких игровых проектов, большой любитель Pascal и почитатель C++. На данный момент, работает в команде над очередной игрой, а в свободное время поддерживает сайт linuxmedia.ru.

В жизни любого игромана наступает волнующий момент, когда он вдруг осознает собственное величие и начинает мечтать о создании грандиозной игры всех времен и народов. Как правило, после долгих месяцев разработки, большого количества выпитого пива и выкуренных сигарет выясняется, что все, что делалось, было изначально неверно или привело совсем не к тому результату. После этого молодая команда распадается, проникаясь неверием в свои силы и вообще в звезду гейм-дева. Целью данной статьи является предостережение от неверных шагов и небольшая порция советов, основанных на личном опыте.

Итак, с чего начинается игра? Разумеется, с идеи! А идей бывает очень много, особенно на начальном этапе. Необходимо выделить суть будущей игры и попытаться законспектировать ее членораздельно для понимания всеми участниками проекта.

Особый вопрос стоит о взаимоотношениях внутри команды. Не секрет, что основой работы молодых команд является энтузиазм, желание создать «игру своей мечты». Как правило, ее члены объединены какой-то общей идеей, ради воплощения которой, они, собственно, и скооперировались. В обсуждениях, и часто весьма жарких, привносятся новые и подчас сногшибательные «фичи», причем каждый хочет сделать именно по-своему. При этом нередко происходит дикое смешение различных жанров.

Совет №1. Забудьте об «игре своей мечты»! Если вы хотите добиться реальных результатов, остановитесь на чем-нибудь конкретном, пусть это весьма отдаленно напоминает первоначальный замысел.

Очень важно на начальном этапе ввести некое подобие субординации между участниками. Понятное дело, что все тут собрались одаренные, работают на голом энтузиазме, но тем не менее должны быть люди, ответственные за организационную часть. Если у вас, к примеру, есть гейм-дизайнер, то за ним решающее слово о введении в игру каких-либо новшеств. И, обязательно, должен быть один руководитель проекта, взваливший на себя решение основных вопросов (он вполне может совмещать несколько должностей: скажем, координатора и программиста). Очень желательно наличие в команде всех необходимых специалистов: отсутствие какого-либо звена влечет за собой излишний простой и, вследствие этого, остывание к игре всех остальных.

В основе любой мало-мальски серьезной игры лежит документ, имеющий характерное название – «дизайн-документ», скрупулезно описывающий игровой мир проекта (характеристика главного героя, NPC, расы, оружие, ход событий и т.д.). Может, это и кажется излишним бюрократизмом, но вернитесь к тому моменту, когда говорилось об индивидуальном видении идеи. Человек имеет тенденцию забывать обо всем, кроме текущей задачи. Впоследствии этот документ предоставляется издателю; его отсутствие заявляет, мягко говоря, о несерьезном отношении разработчиков к своему детищу. Дизайн-документ – это паспорт игры!

Совет №2. Не поленитесь и разложите по полочкам документально то, что вы хотели бы видеть в игре. Каждый в команде должен знать, над чем он работает и во что это выльется.

Следующей крупной ошибкой является желание всенепременно использовать свой движок. Ясно дело, ваш код самый лучший и будет иметь совсем уж запредельные возможности, но прикиньте, сколько времени вы потратите на его создание, отладку, тестирование… а как же игра? Совсем немногие компании имеют возможность заняться разработкой собственного движка. Чаще всего берется уже готовый код и оптимизируется под свои нужды. Под Linux имеется немало интересных (и главное, бесплатных!) разработок, которые, возможно, вам приглянутся: *SDL*, *Ogre*, *Nebula*, *GLScene*, *QuakeEngine I*, *II*...

Если вы предполагаете распространять игру на коммерческой основе – а это вполне естественное желание – следите за лицензионной чистотой. Издатель обязательно поинтересуется программным обеспечением, которое вы использовали. К счастью, Linux имеет немало уникальных инструментов, созданных на основе бесплатных лицензий. К примеру, если ваш 3D-моделлер желает использовать *3DS Max*, намекните ему о существовании не менее мощного и бесплатного редактора *Blender* (кстати, *Blender* имеет прямой экспорт модели в формате движка *Ogre*).

И, наконец, на форумах, посвященных гейм-деву, часто можно встретить вопрос: «А какой язык программирования мне использовать?». Да какой знаете, тот и используйте. Но! Следует помнить о том, что назначения и возможности у языков различные. Так, C# или Pascal хороши для разработки дополнительного инструментария, Pyhton или Lua – для скриптов, а вот основной код лучше все-таки писать на C/C++. Тем более, некоторые движки имеют привязки к определенным языкам. Между прочим, планируйте свои ресурсы с учетом того, что придется писать немало промежуточных утилит: всевозможных редакторов частиц, уровней, файлов конфигурации, и т.п. Вот для таких задач очень даже подойдут быстрые средства разработки.

Совет №3. Тщательно планируйте программную сторону проекта с учетом ваших возможностей. Не стоит, скажем, тратить силы на создание уровней в Blender, лучше написать специализированный редактор для этих целей. В то же время, создание дополнительного инструментария не является первоочередной задачей, и если есть возможность найти что-то готовое и подходящее – не идите на поводу своих амбиций, а используйте уже имеющееся.

Подытожим. Создание игры – процесс длительный, увлекательный и достаточно затратный. Средств для этого, ясное дело, у начинающих не имеется, поэтому важно все время подогревать интерес людей к проекту. Если человек не видит результата своей работы, то он очень быстро охладевает к ней.

Старайтесь шире использовать чужой код с целью экономии времени. В Интернете можно найти немало готовых разработок, к примеру, на www.sourceforge.net. В этом нет ничего зазорного!

Следите за лицензионной чистотой, заранее думайте о кислом выражении лица издателя. Это касается не только инструментария разработки, но и основного кода, арта, музыки.

Ну и удачи, и большо-о-о-го терпения! LXF

Игры осваивают GNU/Linux

Мы решили, что первый выпуск Игрострой LXF будет неполным без интервью с человеком, который уже сейчас разрабатывает игры для Linux. Поэтому сегодня **Евгений Балдин** беседует с Михаилом Георгиевичем Куринным – «отцом» *AstroMenace*.

Одна из причин – правда, не из основных – по которой я полностью перешел на GNU/Linux, это абсолютное отсутствие игр. Тогда я действительно так думал. Но время идет, и «игроделы» начинают осваивают и эту новую для себя среду. Зачем они это делают?

Михаил Георгиевич Куринной возглавляет небольшую компанию Viewizard Games (http://www.viewizard.com), цель которой – делать игры. Он не только самый большой начальник в своей компании, но и программист, и «гейм-дизайнер». Создается ощущение, что Михаил все это устроил только для того, чтобы делать игры, в которые ему самому хочется играть, благо первое техническое и второе экономическое образование ему это позволяют. Интересно узнать у образованного человека, зачем он выпускает свою игру под GNU/Linux бесплатно.

Linux Format: Почему вы начали заниматься созданием игр?

Михаил Г. Куринной: Все начиналось банально, так же, как и у многих. Однажды, играя в свою любимую игру *DemonStar* (космическая скроллинговая стрелялка от *Mountain King Studios*), я подумал, что неплохо было бы кое-что добавить, а кое-что изменить. С интересом начал перекапывать Интернет, одновременно пытаясь сделать хоть что-то похожее на любимую игру. Поначалу ничто не предвещало «трагедии», и у всех была надежда, что у меня это пройдет, что это возрастное [улыбается].

LXF: Ваша основная на сегодня игра *AstroMenace* (см. LXF95) продается для пользователей Windows, а для пользователей GNU/Linux доступна бесплатно. Чем объясняется такая политика?

МГК: Эта политика объясняется тем, что я так захотел с самого начала. В любом предприятии директор/хозяин/главный на свой страх и риск вынужден принимать различные решения, вот это и было одно из них.

Сейчас я могу привести огромное количество «оправданий», сказать, что мы получили от Linux-сообщества даже больше, чем ожидали, что линуксоиды обнаружили и помогли устранить порядка 90% всех обнаруженных ошибок, сделали массу предложений по улучшению игры и помогают с переводами на другие языки, тем самым значительно сокращая наши расходы и косвенно поднимая продажи платной версии игры. Конечно, все это так, но ведь когда решение принималось, я всего этого не знал.

LXF: По какой причине вы заинтересовались GNU/Linux?

МГК: GNU/Linux меня заинтересовал примерно с 2004 года, но поставить его к себе на компьютер и вплотную заняться изучением удалось только в марте 2006. Через 2 месяца я уже смог полностью отказаться от использования другой операционной системы, я думаю, вы поняли, какой [улыбается].

Что больше всего в GNU/Linux мне нравится как пользователю? Да, пожалуй, все. Как-то сразу система пришлась по душе. Как разработчик игр, GNU/Linux я оценил буквально в течение первой недели «адаптации». Тут вам и полный набор всевозможных средств разработки, и огромный выбор IDE, как говорится, на любой вкус и цвет. Есть и хороший выбор уже готовых кросс-платформенных библиотек, способных без проблем заменить DirectX.

LXF: Ваша модель бизнеса идентична shareware. Насколько сейчас это перспективно? Проектов с такой моделью продаж, ориентированных на Linux, фактически нет.

МГК: На самом деле в Linux эта модель только набирает обороты, так как GNU/Linux только выходит на рабочие столы (целенаправленно примерно с 2002–2003 годов, в то время как Windows-«шаровара» зарождалась аж в 1989–1990), так что она может быть не очень актуальна сейчас, но очень перспективна.

Я лично видел уже несколько игрушек, причем одна, что называется, «выстрелила», и разработчики писали об отличных продажах.

LXF: Какова ваша точка зрения на перспективы распространения игр под GNU/Linux в обозримом будущем?

МГК: Я думаю, в ближайшее время нас ждет дальнейшая борьба настольных дистрибутивов GNU/Linux за право по крайней мере на «dual boot». Очень надеюсь, игра *AstroMenace* в этом хотя бы немного поможет. В будущем, по моему мнению, все выглядит весьма перспективно. Все больше компаний понимают, что линуксоиды – тоже люди, ой, в смысле, тоже покупают программное обеспечение [улыбается].

LXF: Каковы ваши планы в отношении игр и GNU/Linux?

МГК: Сейчас основное направление – это развитие игр серии *AstroMenace*. Основной упор делается и будет делаться на использование последних достижений в программном и аппаратном обеспечении. Хочется, чтобы, посмотрев на игру, запущенную под Linux, человек сразу понял – это, вот именно это, настоящая игровая операционная система, и он явно что-то в жизни пропустил и надо срочно наверстывать.

LXF: Чего не хватает в GNU/Linux лично вам для вашей непосредственной работы?

МГК: Вы знаете, самое смешное, что над этим вопросом я думал дольше всего. У меня все есть, и все работает… даже не знаю, чего и пожелать.

LXF: Тогда чего бы вы хотели пожелать сообществу Open Source?

МГК: Хотелось бы пожелать много новых интересных идей и их успешной реализации, а также, чтобы меньше отвлекали от работы всевозможными юридическими и финансовыми аспектами [улыбается]. LXF

ASTROMENACE

Движки для игр

«Что наша жизнь? Игра…» А сердце любой игры заключено в той части кода, которая в простонародье кличется «движком». **Андрей Прахов** заглянул прямо в сердце…

Прежде, чем мы начнем повествование, необходимо четко разъяснить то, что понимается под вынесенным в заголовок термином. Очень часто, говоря о движке, люди связывают его с определенной игрой. Это не совсем верно. Технологии, используемые, скажем, в *DOOM 3*, вполне могут быть лицензированы третьей фирмой. И не факт, что получившаяся у них игра затмит оригинал или даже задействует все крутые разработки конкретного движка. Кроме того, движок – это не только исполняемый код ядра, но и его окружение в виде вспомогательных утилит и, что немаловажно, поддержка производителя.

Возьмем гипотетическую компанию «XGames», решившую создать очередной шедевр. Пусть в свою игру они предполагают включить зрелищные поединки на мечах. Что они могут предпринять? В принципе, путей только три: лицензировать чужую технологию, написать свой код, воспользоваться разработками Open Source. Однако какой бы путь они ни выбрали и какие бы запредельные возможности приобретенная ими система «файтинга» ни имела – если нет сопутствующего инструментария для воссоздания задуманных движений и эффектов, то львиная доля возможностей системы будет просто не задействована.

Хороший движок – это не монолитное сооружение, а скорее конструктор, призванный воплотить цели разработчика. В принципе, можно выделить неизменяемую часть – «Core Engine» – и систему подключаемых модулей. Именно от качества кода основного ядра и имеющихся модулей зависит диапазон возможностей игр, которые теоретически можно реализовать на данном движке.

Существующие движки можно разбить по категориям: графические, физические, скриптовые, звуковые, AI [искусственный интеллект, – прим. ред.] и т.д. Нет в мире универсального движка, решающего все эти задачи. Скажем, *GLScene* имеет в своем составе физическую библиотеку *ODE*, а *Unreal Engine II* слит с *Havok*. Поэтому далее мы рассматривать будем отдельных представителей конкретных направлений. Кроме того, только последователей открытости и бесплатности. И этому есть как минимум пара причин. Во-первых, стоимость графического движка колеблется от $5000 до бесконечности. Так, в свое время, *Source* продавался почти за миллион долларов. Цены на физические чуть поскромнее: славный представитель этого семейства *Havok*, по слухам, стоит около $50000 (всего-то!). И откуда брать деньги разработчикам, начинающим и не очень? Опять-таки, качество отдельных бесплатных разработок заслуживает уважения, и не раз они использовались в коммерческих проектах.

При выборе движка нужно учитывать специфику текущего проекта. Так, *Quake Engine II* хорош для прорисовки закрытых пространств и весьма непродуктивен на открытых. Но есть и универсальные движки. К таким можно отнести *GLScene*.

GLScene

Когда-то разработанный для использования в среде Delphi (Windows), он благополучно перекочевал и на Linux. Работать с ним можно либо из-под *Lazarus* (открытый аналог *Delphi*), либо из-под *FreePascal*. Понятное дело, что в основе его лежит работа с OpenGL и язык программирования Object Pascal. Это вполне шустрый и современный движок, технически не отстающий от веяний времени. Подчас разработчикам удается воплощать новое быстрее, чем соответствующее оборудование появляется на массовом рынке. Помимо графической части, движок имеет

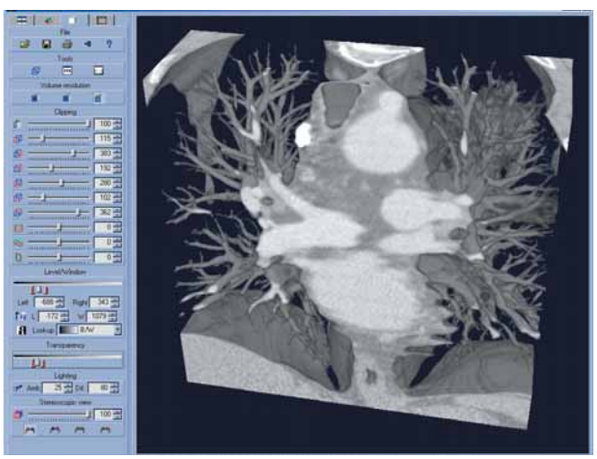


Рис. 1. GLScene: Medical Volume Explorer

в своем составе модули физики (*ODE*, *DDE*), умеет работать со звуком и сетью, есть встроенный редактор сцены (правда, очень и очень примитивный). Конечно же, поддерживаются шейдеры, скелетная и покадровая анимация, частицы, motion blur, bump-mapping и многое другое. Помимо своего непосредственного назначения (работы с трехмерной графикой), *GLScene* неплохо умеет обращаться и с двумерными спрайтами, примитивами. Для разработки интерфейса предназначен специальный класс, что существенно облегчает жизнь программиста, а имеющиеся в наличии небольшие утилиты освобождают от рутинности разработки. Большой плюс – наличие документации, примеров и целой армии поклонников.

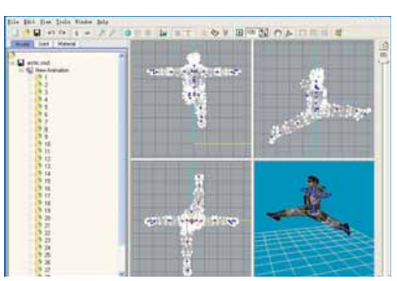


Рис. 2. GLScene: Game3D-Max

Особенности движка – серьезная продуманность внутренней структуры и большое количество встроенных заготовок. Так, создать полноценное небо или водную поверхность можно буквально несколькими командами. Не обойден стороной и звук: работать с ним – одно удовольствие. Естественно, поддерживается трехмерное окружение.

Нельзя сказать, что это очень быстрый движок. Да, стрелялку класса AAA на нем не сделаешь, но для небольших казуальных игр или не слишком динамичных RPG он в самый раз. По секрету: разработчики *GLScene* никогда не позиционировали его только лишь для программирования игр.

Рис. 3. GLScene: Virtual studio

Рис. 4. OGRE: отражения

OGRE

Совсем недавно завершился цикл статей по программированию игры на *OGRE* в нашем журнале, поэтому остановимся на нем вкратце. Читатели наверняка попробовали на ощупь мощь и элегантность работы движка. Созданный аж в 2002 году, *OGRE* активно разрабатывается и поныне, заслуженно считаясь наиболее динамично развивающимся и перспективным проектом такого рода. Продуманная архитектура, богатый набор документации и примеров, большое количество сопроводительного инструментария – все это делает движок интересным не только для простых пользователей, но и для коммерческих проектов (примером может служить игра *«Стальные монстры (Pacific Storm)»* студии «Леста»). Кроме того, по слухам, просочившимся из стана разработчиков *Blender*, этот движок скоро ляжет в основу следующих версий редактора.

Итак, что же «вкусного» он может нам предложить? Во-первых, независимость от типа графического API, позволяющую с легкостью переключаться между DirectX и OpenGL. Поддержка трех операционных систем – Linux, Windows, Mac OS X. Богатый набор модулей расширения для работы с различными форматами файлов трехмерных редакторов (*3ds Max*, *Maya*, *Blender*, *MilkShape*…). Движок написан на языке программирования C++.

Рис. 5. OGRE: игра Heretic Kingdoms

И хотя он позиционируется в первую очередь как графический движок, остальные стороны игростроя не остались в стороне. Так, имеется пример использования открытой физической библиотеки ODE совместно с *OGRE*.

Итого: мощная и открытая архитектура, солидная документация, технологичный язык программирования – все это делает *OGRE* привлекательным и для коммерческого использования.

The Nebula Device

История этого проекта достаточно необычна и поучительна. Немецкая студия «Radon Labs» при создании игры *«Project Nomads»* встала перед обычной дилеммой: то ли лицензировать чужой движок, то ли написать свой собственный. По некоторым причинам, движки, предлагаемые рынком, не подходили для задуманного, но и ресурсов для создания своего явно не хватало. Ребята поступили просто и мудро – открыли свои наработки, и впоследствии, не без помощи сторонников Open Source, получилась красивая и изящная игра.

Рис. 6. OGRE: виртуальный музей. Нант 18-го века.

Движок этого типа призван для обработки исключительно открытых пространств. Изумительное по красоте небо, реальные облака, детализированные острова и видимость на десятки километров – вот что предложила игрокам эта игра, а соответственно, и ее движок.

Подобно *OGRE*, движок является мультиплатформенным, он поддерживает DirectX и OpenGL. Как и его вышеописанные собратья, он предлагает разработчикам мощь современных технологий и легкость (относительную) программирования.

Написан на C++, поддерживает несколько скриптовых языков (в том числе и любимый разработчиками игр Lua). Имеется даже возможность включить поддержку иного языка с помощью системы расширений.

Работа с различными форматами графических файлов также сделана на подобающем уровне. Движок «понимает» форматы BMP, JPG, GIF, TIFF, PNG и некоторые другие. Исключение составляет поддержка файлов трехмерных редакторов. Помимо собственных форматов NVX и N3D, можно использовать лишь OBJ (слабое место!).

В своем составе имеет несколько дополнительных утилит. Так, к примеру, для контролирования источников света можно воспользоваться специальным инструментом (Light Control Tool).

Ну и, под конец, еще одна большая ложка дегтя – почти полное отсутствие документации. Впрочем, на собственном сайте Nebula имеются кое-какие примеры и руководства.

Irrlicht

Завершает повествование о графических движках обзор еще одного интересного проекта. Из всех рассмотренных, только *Irrlicht* имеет возможность использовать программный режим отрисовки. Скажете, древность? А вот и нет. Представьте, что для разработки небольшой казуальной игры вы воспользовались, скажем, тем же *OGRE*. Получилось красиво, быстро, интересно. Но вот беда, на обычных конторских машинах не всегда стоят видеоплаты с 3D-ускорителями (или соответствующие драйвера), а игры такого класса рассчитаны по большей степени именно на них. Потеря потенциальных потребителей – дело неприятное. *Irrlicht* же позволяет с легкостью переключаться с аппаратного ускорения на программный рендер.

Движок достаточно новый. Не совсем еще стабильный, но основные функции для разработки уже имеются. Набор предлагаемых эффектов достаточно богат: создание реалистичных водных поверхностей, динамическое освещение, световые карты, динамические тени, работа с биллбордами (billboards), система частиц и многое другое.

Irrlicht, понятное дело, является кроссплатформенным (Windows, Linux, MacOS, Sun Solaris/SPARC), написан на C++, понимает форматы OBJ, 3DS, MD2 (формат *Quake Engine*). Легко интегрируется с физическими библиотеками *ODE*, *Tokamak*.

…и другие

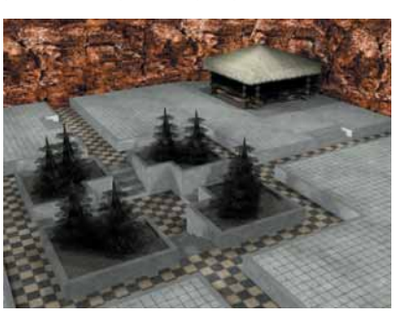
Рис. 7. Irrlicht: игровой движок RABCAT

› Рис. 8. Irrlicht: фикшн-гонки H-Craft Championship

Конечно же, количество существующих графических движков, подходящих для использования в Linux, не ограничивается лишь четырьмя ранее рассмотренными. Существует масса узкоспециализированных, разработанных для создания игр разного жанра. Но есть еще одна категория движков, без которых не обходится ни одна из современных игр – а именно, физические.

Уже из названия понятно их назначение. Столкновения, падения, гравитация, зрелищные взрывы – все, что основано на законах физики, находится во власти этого кода.

Не стоит думать, что эти расчеты достоверно моделируют физические процессы на все 100%: как правило, происходит лишь аппроксимация. Расчет в реальном времени требует немалых процессорных ресурсов, и поэтому разработчики идут на всевозможные ухищрения, типа разбиения игрового пространства на обсчитываемые зоны, нестандартного использования GPU высокоскоростных видеоплат и, наконец, использования специализированных физических ускорителей.



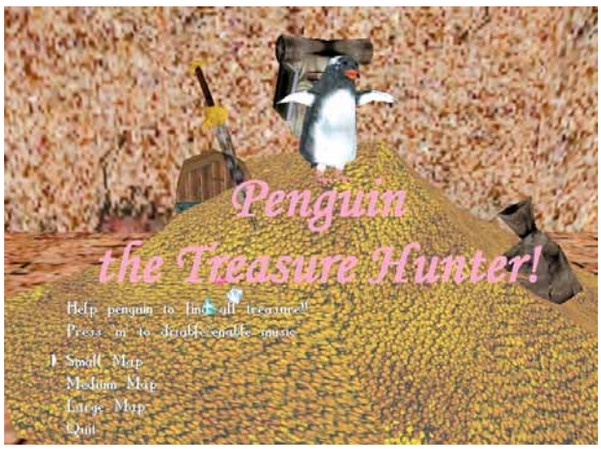
› Рис. 9. Irrlicht: RPG IrrWorld

Примером последнего может служить движок *PhysX* компании Ageia. Это пока единственный движок, способный использовать всю мощь PPU (Physics Processing Unit); кроме того, он самый шустрый. Но есть и погрешности моделирования. Имеется несколько бесплатных лицензий (без поддержки PPU) для коммерческого использования. По слухам, скоро появятся двоичные пакеты и для Linux.

Гораздо более привычным для разработчиков является движок ODE – старейший представитель, изначально создававшийся под эгидой Open Source, не самый быстрый и, как ни странно, не очень стабильный. Он разрабатывается и поныне, изредка радуя почитателей некоторыми новыми функциями. Умеет просчитывать столкновения не только между стандартными примитивами (сфера, плоскость, куб), но и работает непосредственно с mesh-объектами. Используется имитация физики твердого тела (Rigid body sumulation), на основе которой можно воссоздать широко известный эффект «тряпочной куклы» (ragdoll). Является стандартом de-facto использования совместно с другими графическими движками, такими, как *OGRE*, *GLScene*, *Nebula*. Из особенностей: богатая документация, несложная и понятная реализация функций. Лучше всего проявляет себя при обсчете простых физических тел в небольшом проекте. Однако, благодаря своей открытости и документированности, может стать основой для разработки более сложного кода (появилось желание создать конкурента *Havok*?).

› Рис. 10. Irrlicht: космическая стратегия Galactic Dream

Напоследок хочется дать добрый совет: не гонитесь за самым мощным движком, лучше используйте простой, но хорошо документированный! LXF



› Рис. 11. Irrlicht: бродилка Penguin, the Treasure Hunter!

Справочник игродела

» **Billboards** Двухмерные спрайты, всегда повернутые «лицом» к активной камере. Часто используются для создания объектов (или подмены «реальных» трехмерных объектов), находящихся на удалении от камеры, с целью уменьшения нагрузки на систему.

» **Bump-Mapping** Технология выдавливания для придания шероховатости поверхности объекта. Неровности создаются одной картинкой (бамп-карта), состоящей из градаций серого цвета (от 0 до 255). Чем светлее пиксель, тем выше он находится от поверхности. Кроме него, имеются более современные способы: Normal Mapping, Parallax Mapping, Relief Mapping.

» **Collision Detection** Обнаружение столкновений между объектами с последующей реакцией программы. Расчеты могут вестись либо на уровне примитивов (сфера, куб), либо на уровне граней (mesh) самого объекта. В последнем случае точность определения значительно возрастает за счет увеличения нагрузки на систему.

» **Motion Blur** Эффект размытости изображения при движении. Как известно, при высокой скорости движения объекта в реальном мире его очертания расплываются. Данная технология придаст более зрелищный вид любой трехмерной игре.

» **Particles System** Предназначена для моделирования объекта с неопределенной формой. В свою очередь, сам объект состоит из подмножества других, каждый из которых рождается, живет и умирает в конкретный момент, по замыслу аниматора. Частицы могут порождать другие частицы, приобретать новые свойства и перемещаться по заданным траекториям. Таким образом, контролируя цвет, прозрачность, размер и динамику поведения частиц, можно получить различные визуальные эффекты: например, облака, пламя, водные брызги, дым.

› Рис. 12. Irrlicht: скринсейвер MyPictures 3D

VII МЕЖДУНАРОДНАЯ ВЫСТАВКА-ФОРУМ

ИнфоКом-2007

инфокоммуникации России - XXI век

при поддержке Министерства информационных технологий и связи Российской Федерации

24 - 27 ОКТЯБРЯ 2007 ГОДА
Москва, МВЦ „Крокус Экспо"

Разделы экспозиции:

Информационные технологии

- Программное обеспечение
- Системная интеграция
- Информационная безопасность

Телекоммуникации

- Проводная (фиксированная) связь
- Беспроводная (мобильная) связь
- Инфокоммуникационные услуги

Почтовая связь

Специальные тематические экспозиции

- Национальные проекты
- Технопарки
- Инновации
- Электронное правительство
- Электронная культура
- Интеллектуальный дом

Дирекция выставки :

Тел. : +7 (495) 234-50-10
Факс: +7 (495) 234-50-09
www.infocomtech.ru

Телефон горячей линии:
8-800-333-9-333

Премьер-партнеры :

СВЯЗЬ ИНВЕСТ

СИСТЕМА ТЕЛЕКОМ

Информационные партнеры:

LINUX FORMAT
Главное в мире Linux

Журнал зарегистрирован Федеральной службой по надзору за соблюдением законодательства в сфере массовых коммуникаций и охране культурного наследия
ПИ № ФС77-21973 от 14 сентября 2005 года
Выходит ежемесячно. Тираж 6000 экз.

РЕДАКЦИЯ РУССКОЯЗЫЧНОЙ ВЕРСИИ:
ГЛАВНЫЙ РЕДАКТОР
Валентин Синицын info@linuxformat.ru
Литературные редакторы
Родион Водейко, Елена Толстякова, Иван Мищенко
Переводчики
Илья Аввакумов, Регина Бабыкина, Александр Бикмеев, Светлана Кривошеина, Александр Казанцев, Алексей Опарин, Валентин Развозжаев, Юлия Шабунио
Редактор диска
Александр Кузьменков
Допечатная подготовка
Мария Пучкова, Родион Водейко
Креативный директор
Станислав Медведев
Технический директор
Денис Филиппов
Директор по рекламе
Денис Игнатов +7 812 965 7236 advert@linuxformat.ru
Заместитель генерального директора
Софья Виниченко
Генеральный директор
Павел Фролов
УЧРЕДИТЕЛИ
частные лица
ИЗДАТЕЛИ
Станислав Медведев, Виктор Федосеев, Павел Фролов

Отпечатано в типографии «Текст», ООО «ППК «Текст»
188680, Ленинградская область, Всеволожский район, Колтуши, д.32
Заказ _____

Пре–пресс: *d.r.i.v.e-group*

РЕДАКЦИЯ АНГЛОЯЗЫЧНОЙ ВЕРСИИ:
Редактор Ник Вейч (Nick Veitch) nick.veitch@futurenet.co.uk
Заместитель редактора Пол Хадсон (Paul Hudson) paul.hudson@futurenet.co.uk
Художественный редактор Эфрейн Эрнандес-Мендоса (Efrain Hernandez-Mendoza) efrain.hernandez-mendoza@futurenet.co.uk
Новостной редактор Майк Сондерс (Mike Saunders) mike.saunders@futurenet.co.uk
Редактор обзоров
Грэм Моррисон (Graham Morrison) graham.morrison@futurenet.co.uk
Литературный редактор
Ребекка Смолли (Rebecca Smalley) rebecca.smalley@futurenet.co.uk
Ассистент по выпуску
Шарлотта Мартин (Charlotte Martyn), Сара Гейнз (Sarah Gaines)

Подготовка материалов
Ладислав Боднар (Ladislav Bodnar), Нейл Ботвик (Neil Bothwick), Д-р Крис Браун (Dr. Chris Brown), Энди Ченнел (Andy Channelle), Крис Хауэллз (Chris Howells), Энди Хадсон (Andy Hudson), Рейчел Проберт (Rachel Probert), Евгений Балдин, Александр Бабаев, Андрей Боровский, Владимир Иванов, Галина Пожарина, Александр Поносов, Андрей Прахов, Петр Семилетов, Сергей Супрунов, Алексей Федорчук, Александр Чернышов, Светлана Шапошникова, Илья Шпаньков.
Художественные ассистенты: Ники Готобед (Nicky Gotobed), Роберт Хейл (Robert Hale), Тина Гленкросс (Tina Glencross), Ник Кокс (Nick Cox), Дэвид Диас (David Dyas)
Фотографии: Джена Камбо (Jena Cumbo)
Иллюстрации: Нейл Бартлет (Neil Bartlett), Пол Блехфорд (Paul Blachford), Крис Винн (Chris Winn)

КОНТАКТНАЯ ИНФОРМАЦИЯ
UK: Linux Format, 30 Monmouth Street, Bath BA1 2BW
Tel 01225 442244 Email: linuxformat@futurenet.co.uk
РОССИЯ:
Санкт-Петербург (редакция): ул. Гончарная, 23, офис 54, телефон: (812) 717-00-37
Представительство в Москве:
пр.Мира, 161, телефон +7(495) 799-18-63, +7(495)136-88-45
Email: info@linuxformat.ru, Web: www.linuxformat.ru

Авторские права: Статьи, переведенные из английского издания Linux Format, являются собственностью или лицензией Future Publishing Ltd (Future plc group company). Все права зарегистрированы. Никакая часть данного журнала не может быть повторно опубликована без письменного разрешения издателя.

Все письма, независимо от способа отправки, считаются предназначенными для публикации, если иное не указано явно. Редакция оставляет за собой право корректировать присланные письма и другие материалы. Редакция Linux Format получает неэксклюзивное право на публикацию и лицензирование всех присланных материалов, если не было оговорено иное. Linux Format стремится оставлять уведомление об авторских правах всюду, где это возможно. Свяжитесь с нами, если мы не упомянули вас как автора предложенных вами материалов и мы постараемся исправить эту ошибку. Редакция Linux Format не несет ответственности за опечатки.

Все присланные материалы могут быть помещены на CD или DVD-диски, поставляемые вместе с журналом, если не было оговорено иное.

Ограничение ответственности: используйте все советы на свой страх и риск. Ни при каких условиях редакция Linux Format не несет ответственность за повреждения или ущерб, нанесенные вашему компьютеру и периферии вследствие использования тех или иных советов.

За содержание рекламных материалов редакция ответственности не несет.

Linux-зарегистрированная торговая марка Линуса Торвальдса (Linus Torvalds). Название «GNU/Linux» заменяется на «Linux» в целях сокращения. Остальные торговые марки являются собственностью их законных владельцев.
Linux Format является торговой маркой Future Publishing Ltd (Future plc group company).

За информацией о журналах, издаваемых Future plc group company, обращайтесь http://www.futureplc.com

© Linux Format 2005
© Future Publishing Ltd 2005

BATH • LONDON • MILAN • NEW YORK • PARIS • SAN DIEGO • SAN FRANCISCO

в октябре

Создавая Ubuntu

Что вкладывается в понятие «выпускать дистрибутив мирового уровня»? Мы взглянем на разработку, тестирование, художественное оформление и процедуру контроля качества, без которых невозможен релиз Linux...

Сведи на нет Slashdot-эффект

Ваш *Apache* задыхается уже при двух запросах в секунду? Узнайте, как сделать его сверхскоростным!

Сколько стоит свобода?

Грэм Моррисон ценит свободу, но он также ценит и работающие инструменты.

Луи Поттс

Что ждет пользователей *OpenOffice.org* в 2008 году?

Содержание последующих выпусков может меняться без уведомления